



Universidad
Zaragoza

Trabajo Fin de Grado

Diseño de juegos tangibles para Espacios Interactivos, orientados a formar a niños en el pensamiento lógico y computacional

Design of Tangible video-games in an Interactive Environment, aimed for training logical and computational reasoning to children

Autora

Ester Ruiz Figueroa

Director

Dr. Javier Marco Rubio

Ponente

Dra. Eva Cerezo Bagdasari

ESCUELA DE INGENIERÍA Y ARQUITECTURA

2017



DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe acompañar al Trabajo Fin de Grado (TFG)/Trabajo Fin de Máster (TFM) cuando sea depositado para su evaluación).

D./D^a. Ester Ruiz Figueroa,

con nº de DNI 73222635L en aplicación de lo dispuesto en el art.

14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo

de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la

Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de (Grado/Máster)
Grado _____, (Título del Trabajo)

Diseño de juegos tangibles para Espacios Interactivos, orientados a formar a
niños en el pensamiento lógico y computacional

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada
debidamente.

Zaragoza, 30 de agosto de 2017

Fdo: Ester Ruiz Figueroa

Agradecimientos.

A Javier Marco, por haberme enseñado todo lo necesario para la realización del proyecto y haberme guiado y ayudado durante todo el trabajo. También por ayudarme con la redacción de la memoria y corregir cada una de sus versiones.

A Eva Cerezo, por aceptarme para este proyecto y por abrirme la vista a un mundo de posibilidades en el ámbito de la interacción persona-ordenador.

A Belén Cebrián, ya que sin su ayuda y disposición este proyecto no hubiera llegado tan lejos. Gracias por hacer de Star Loop una realidad.

A los niños que participaron en las diferentes sesiones, sin los cuales no se podría haber probado en unas condiciones adecuadas ni se hubieran alcanzado unas conclusiones reales. Este proyecto no hubiera sido lo mismo sin esta experiencia.

A toda mi familia y amigos, por apoyarme durante esta etapa de mi vida.

Y en especial a mi hermana Marta, por toda su ayuda y por darle voz y vida a Spowi.

Gracias a todos.

RESUMEN

Este trabajo lleva como título “Diseño de juegos tangibles para Espacios Interactivos, orientados a formar a niños en el pensamiento lógico y computacional” y se ha desarrollado en el seno del grupo de investigación GIGA (Grupo de Informática Gráfica Avanzada) Affective Lab del departamento de Informática e Ingeniería de Sistemas de la Universidad de Zaragoza.

Este grupo dispone, gracias a los proyectos CeSAr y JUGUEMOS (proyecto financiado por el ministerio de Educacion Ciencia y Tecnología TIN2015-67149-C3-1R.), un espacio interactivo en el edificio Etopia de Zaragoza (Centro de Arte y Tecnología de Zaragoza) orientado al diseño y desarrollo de juegos pervasivos para niños.

El objetivo general de este proyecto ha sido diseñar juegos pervasivos para el espacio JUGUEMOS con los que enseñar conceptos básicos de programación a niños en un espacio colaborativo y de manera lúdica.

Para la realización de este proyecto se ha partido de una fase analítica en la cual se ha recopilado y analizado toda la información necesaria sobre los niños, los juegos y la actualidad del mercado en lo referente a interacción tangible y programación.

Gracias a esta documentación se han obtenido unas conclusiones y un punto de partida fundamental para comenzar la fase de ideación y diseño, en la cual se han definido dos conceptos independientes a desarrollar paralelamente, un juego para el espacio JUGUEMOS y una ampliación de un juego ya existente en el mercado.

A partir de ahí se han desarrollado paralelamente ambos juegos en un proceso iterativo hasta obtener los prototipos finales de ambos conceptos.

Por último, una vez creados estos prototipos finales de ambos conceptos se realizan unas evaluaciones de ellos en el espacio JUGUEMOS y en el colegio Los Albares de la Puebla de Alfinden. Gracias a la evaluación con usuarios reales se han detectado fallos y se han propuesto posibles opciones de mejora dejando abierta la opción de mejorar estos juegos en un futuro.

ÍNDICE

1. Introducción	8
1.1. Contexto	9
1.2. Objetivos	11
1.3. Metodología	12
1.4. Planificación y desarrollo temporal	14
2. Fase de análisis	16
2.1. Usuario	17
2.2. Interacción tangible	18
2.3. Tecnologías para la enseñanza del pensamiento computacional en niños	19
2.4. Conclusiones fase análisis	24
3. Fase de creación y diseño	25
3.1. Ideación	26
3.2. Definición de los juegos	27
3.2.1. JUEGO 1: Juego para el espacio JUGUEMOS	27
3.2.2. JUEGO 2: Ampliación de TurTan para NIKVision redonda	32
4. Fase de desarrollo e implementación	34
4.1. Star Loop	35
4.2. Ampliación TurTan	52
5. Fase de evaluación y resultados	55
5.1. Star Loop	56
5.2. Ampliación TurTan	60
6. Conclusiones y trabajo futuro	64
6.1. Conclusiones	65
6.2. Trabajo Futuro	66
Bibliografía	67
Anexos	73
Anexo A - Espacio JUGUEMOS	74
A.1. Equipamiento espacio JUGUEMOS	75
A.2. Tabletop NIKVision	77
Anexo B - Usuario	80
B.1. Evolución de las capacidades psicomotoras del niño	81
B.2. Desarrollo intelectual del niño	82
B.3. Desarrollo de los juegos a lo largo de la infancia	84
B.4. Beneficios del juego en el desarrollo integral del niño	85
B.5. Aprendizaje cooperativo en niños	86
Anexo C - Ejemplos de interacción tangible	87
Anexo D - Pensamiento lógico computacional en niños	90

D.1. Aprendizaje del pensamiento lógico computacional en niños	91
D.2. La situación actual sobre la enseñanza del pensamiento lógico computacional y programación en niños	93
Anexo E - Propuestas basadas en Scratch	94
Anexo F - Propuestas basadas en Logo	97
Anexo G - Técnicas creativas aplicadas y conceptos	101
G.1. Panel de influencias	102
G.2. Mapas mentales	103
G.3. Brainstorming	105
G.4. Conceptos	108
Anexo H - Star Loop	110
H.1. Restricciones técnicas e información necesaria para desarrollar los recursos	111
H.2. Documento explicativo inicial	114
H.3. Creación de recursos físicos	130
H.4. Documento explicativo final	131
Anexo I - Ampliación TurTan	152
I.1. Propuestas alternativas del diseño gráfico de las fichas	153
I.2. Prototipado de las fichas	155
I.3. Montaje tabletop NIKVision redondo	156
I.4. Calibración tabletop NIKVision redondo	158
Anexo J - Propuestas de mejora para Star Loop	159
J.1. Soluciones a los problemas menores de usabilidad	160
J.2. Soluciones a los problemas colaborativos	161

INTRODUCCIÓN

1

1.1. CONTEXTO

GIGA Affective Lab

Este trabajo de fin de grado ha sido desarrollado en el ámbito del grupo de investigación del departamento de informática y sistemas de la EINA, el GIGA Affective Lab [1]. Este grupo de investigación se centra en la interacción usuario-ordenador.

El trabajo de este grupo de investigación se ha centrado en los últimos años en cuatro temas principales, que son la interacción natural, la humanización virtual, la computación afectiva y la accesibilidad. Y es en el ámbito de investigación y desarrollo de interfaces de usuario tangibles naturales donde se enmarca el presente trabajo.

El GIGA Affective Lab ha desarrollado la mesa tangible NIKVision [2] y juegos para niños con los que ofrecer nuevas formas de entretenimiento y diversión con fines terapéuticos o de aprendizaje, al mismo tiempo que se refuerza en los usuarios la manipulación física y el juego colocalizado en entornos educativos.

Con el objetivo de facilitar la creación de prototipos de juegos tangibles el grupo GIGA está desarrollando ToyVision [3], un conjunto de herramientas de software que tiene como finalidad estrechar la brecha entre los procesos de diseño y prototipado de juegos tangibles para la NIKVision. Con esta herramienta se pretende facilitar y agilizar, además de unir, los procesos de diseño y prototipado.

Espacio JUGUEMOS en Etopia

El GIGA Affective Lab cuenta con un espacio interactivo en el Centro de Arte y Tecnología de Zaragoza, edificio Etopia (Ver Imagen 1). Esta iniciativa del Ayuntamiento de Zaragoza cuenta con un equipamiento de nueva generación diseñado para albergar y promover los proyectos creativos y emprendedores más innovadores dentro del espacio de la Milla Digital. El equipamiento y aplicación de este espacio se enmarca dentro de los proyectos CeSAr y JUGUEMOS [4].

El proyecto CeSAr (Ver Imagen 2) surge de un convenio entre la Universidad de Zaragoza y el Ayuntamiento de Zaragoza con el fin de acercar la ciencia, la creatividad tecnológica y el arte con nuevos medios a la ciudadanía, favorecer el conocimiento colaborativo y consolidar Etopia como centro de producción de proyectos multidisciplinarios. Este proyecto CeSAr cede un espacio en el edificio Etopia con el equipamiento tecnológico necesario para desarrollar juegos tangibles.



Imagen 1 - Logo Etopia



Imagen 2 - Logo Cesar

El proyecto JUGUEMOS (TIN2015-67149-C3-1-R) es un proyecto de investigación nacional

financiado por el ministerio de educación, ciencia y tecnología y tiene el propósito de promover el desarrollo de los juegos pervasivos para espacios interactivos.

El espacio JUGUEMOS (Ver Imagen 3) es un entorno interactivo diseñado como un laboratorio de nuevas formas de interacción natural y tangible, un espacio para la exploración, el diseño y la implementación de juegos pervasivos. Es en este espacio donde tiene lugar el desarrollo del trabajo de fin de grado.



Imagen 3 - Espacio JUGUEMOS

Este entorno está provisto de tres paredes envolventes que rodean a los usuarios en las cuales se proyectan los recursos gráficos. Además, está equipado con los siguientes recursos:

1. Cuatro mesas de interacción tangible NIKVision
2. Tres proyectores
3. Sistemas de localización Ubisense
4. Soporte de interacción gestual Kinect
5. Altavoces
6. Actuadores
7. Focos LED

El equipamiento mencionado permite la interacción entre el usuario y el entorno alcanzando nuevos métodos de interacción más participativos y colaborativos para los usuarios. Esta tecnología permite el seguimiento y ubicación de los usuarios, además de los gestos y acciones de éstos permitiendo una interacción fluida entre los usuarios y el espacio. Este equipamiento se explica detalladamente en el *Anexo A - Espacio JUGUEMOS*, apartado A.1. *Equipamiento espacio JUGUEMOS*.

1.2. OBJETIVOS

El principal objetivo del presente proyecto es el diseño de juegos para el espacio JUGUEMOS. Estos juegos deben estar orientados al aprendizaje del pensamiento lógico y computacional de los niños.

Se quiere conseguir el desarrollo de juegos que permitan un aprendizaje autónomo, creativo, y colaborativo además de gradual, comenzando desde los conocimientos más básicos de la programación hasta el aprendizaje y aplicación de bucles y procedimientos.

El fin es inculcar en los niños el pensamiento computacional que se basa en la resolución de problemas afrontándolos de forma secuencial, organizada y estructurada.

Estos juegos deben basarse en la interacción tangible (manipulación física) ya que ésta logra hacer accesible para los niños el aprendizaje de un lenguaje informático. De esta forma los niños aprovechan las ventajas que ofrece el aprendizaje de un lenguaje informático sin enfrentarse a la tarea de aprender un lenguaje de programación sintáctico, lo cual solventa los problemas que éstos conllevan como por ejemplo los errores de sintaxis.

Uno de los impedimentos de los niños a enfrentarse al aprendizaje de razonamiento computacional, no solo es la propia sintaxis del lenguaje de programación, sino el método de interacción con el ordenador (teclado y ratón), y que es por eso que actualmente se están buscando formas alternativas de interacción con el entorno de programación, más naturales e intuitivas, basadas en la manipulación física, lo cual no solo minimiza los errores de sintaxis, sino que resultan mas atractivas y naturales para los niños, evitando que se aburran y pierdan el interés.

Por otro lado, en su diseño se deben tener en cuenta los recursos tecnológicos disponibles en el espacio JUGUEMOS para su aprovechamiento. También deben permitir la colaboración entre niños fomentando la comunicación e interacción entre éstos.

En consecuencia, los juegos deben cumplir una serie de requisitos:

1. La interacción entre los usuarios y el espacio debe ser física (tangible).
2. Los juegos deben fomentar la colaboración entre los usuarios.
3. El aprendizaje debe ser autónomo y creativo.
4. Los juegos deben permitir jugar hasta a 4 niños al mismo tiempo.
5. Aprovechar los recursos tecnológicos disponibles del espacio JUGUEMOS.
6. El aprendizaje debe ser progresivo, comenzando por retos fáciles y evolucionando poco a poco hasta retos complejos.
7. Se quiere alcanzar el aprendizaje de procedimientos, bucles.

1.3. METODOLOGÍA

Con el fin de alcanzar los objetivos del proyecto descritos anteriormente se lleva a cabo una metodología estructurada en cuatro fases, descritas a continuación.

FASE DE ANÁLISIS

En esta primera fase se realiza una investigación sobre aquellos aspectos que tienen relación con el proyecto a realizar.

Usuario.

Es imprescindible conocer al usuario para diseñar los juegos para éste. Por ello en esta fase se busca todo tipo de información relacionada con los niños, su aprendizaje y su entorno, así como las diferencias y características a lo largo del desarrollo infantil.

Pensamiento lógico computacional.

Debido a que los juegos propuestos deben introducir y guiar a los usuarios hacia un razonamiento lógico característico del pensamiento científico y la programación se busca información acerca de este pensamiento y su aplicación en la enseñanza con un usuario infantil.

Interacción tangible.

Estudiamos todo lo relacionado con la interacción tangible, sus aplicaciones actuales en el mercado y las posibilidades que ofrece.

Productos existentes en el mercado.

Además buscamos todos los productos creados para niños con la intención de introducirlos en el mundo de la programación.

Entorno y recursos disponibles.

Se estudia detalladamente el espacio interactivo y los recursos disponibles con el fin de tenerlos en cuenta para el diseño de los juegos.

FASE DE CREACIÓN Y DISEÑO

Una vez realizada la investigación previa y con toda la información y datos presentes comienza la segunda fase del proyecto.

En esta fase se realizan métodos creativos con el fin de generar conceptos a partir de los cuales se diseñan y definen los juegos. Encontramos técnicas creativas y de selección que nos sirven para crear conceptos y elegir el más adecuado para el diseño final. Una vez elegido el diseño final se define el guión y storyboard de los juegos.

FASE DE DESARROLLO E IMPLEMENTACIÓN

Una vez diseñado el juego se crean todos los recursos gráficos, animaciones, sonidos y todo recurso necesario para la implementación inicial de los prototipos. Tras la primera implementación se realiza un trabajo iterativo (Ver Imagen 4) con el fin de resolver los problemas que se encuentran durante la implementación y prueba de los juegos que no han sido previstos anteriormente. Así, al finalizar esta fase los juegos desarrollados deben estar listos para su evaluación con usuarios finales (niños).

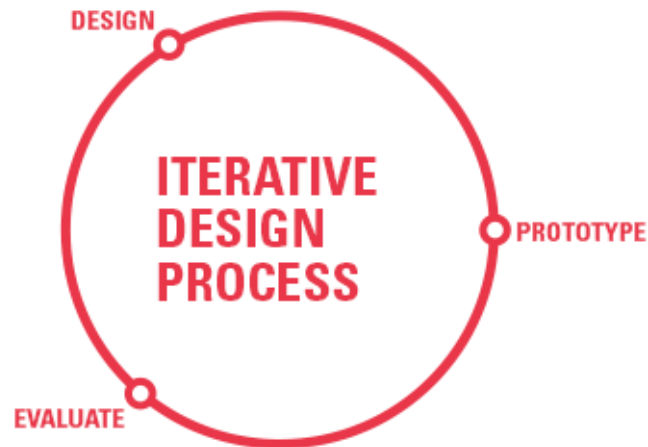


Imagen 4 - Esquema de proceso de diseño iterativo

Cada iteración consiste en el refinamiento, prueba y análisis del prototipo con el fin de resolver los problemas localizados. El refinamiento del prototipo abarca el rediseño y modificación del concepto, de la forma de interacción y los recursos tanto visuales como sonoros. Tras este refinamiento se deben volver a implementar los cambios. Para la implementación se cuenta con una becaria contratada en el proyecto CESAR para el desarrollo informático del prototipo. Y por último el prototipo se evalúa para detectar fallos y si es necesario este proceso iterativo se vuelve a repetir.

Es importante destacar que las iteraciones se evalúan por expertos (adultos), mientras que la evaluación final se realiza con usuarios finales (niños).

FASE DE EVALUACIÓN Y RESULTADOS

En esta fase se evalúan los juegos con usuarios reales. Se observa, analiza y estudia el comportamiento y la experiencia de los niños durante el juego tanto durante las sesiones como con la posterior revisión de los vídeos realizados en las sesiones.

De esta forma se reflexiona acerca de los posibles errores encontrados y se buscan mejoras y soluciones para éstos.

1.4. PLANIFICACIÓN Y DESARROLLO TEMPORAL

Desde el comienzo del trabajo de fin de grado, en Febrero de 2017 se realizó una planificación a lo largo del tiempo del proyecto y se estimó como fecha de finalización Septiembre de 2017. A continuación se muestra esta planificación.

Tabla 1 - Planificación general del Trabajo de fin de grado

	Feb.	Mar.	Abril	May.	Jun.	Julio	Ago.	Sept
Fase de análisis								
Fase de creación y diseño								
Fase de desarrollo e implementación								
Fase de evaluación y resultados								
Redacción de la memoria								
Presentación								

Tabla 2 - Planificación detallada del Trabajo de fin de grado

	Feb.	Marzo	Abril	Mayo	Junio	Julio	Agos.	Sept.
Inicio del trabajo								
Planificación de tareas								
Estudio de los recursos								
Estudio del usuario								
Estudio de mercado (interacción tangible)								
Estudio de mercado (programación y niños)								
Conclusiones fase análisis								
Ideación								
Selección de ideas								
Diseño de juegos								
Propuesta								
Creación y modificación de recursos gráficos								
Creación de juguetes físicos								
Implementación								
Prueba de juegos								
Evaluación de juegos								
Observaciones y reflexión								
Redacción memoria								
Presentación								

FASE DE ANÁLISIS

2

2.1. USUARIO

A la hora de diseñar una aplicación o juego interactivo para un usuario es importante analizar y definir el usuario objetivo para determinar las necesidades y habilidades de éste y realizar un diseño adecuado para él.

Para conseguir un diseño adecuado para un usuario infantil se realiza previamente un estudio de usuario que abarque la información necesaria para comprender las características, capacidades y desarrollo evolutivo de los niños con el fin de determinar un rango concreto de edad sobre el que llevar acabo este proyecto.

En este estudio previo se busca información relativa a la evolución psicomotora de las capacidades del niño de los 0 a los 12 años y el desarrollo intelectual del niño por periodos según Piaget. También se incluye información sobre los beneficios que aporta el juego al desarrollo integral de los niños, el aprendizaje cooperativo y las particularidades metodológicas que le acompañan.

Este estudio se encuentra en el *Anexo B - Usuario*. Gracias a este estudio se puede determinar cual es la mejor edad para relacionarse con otros niños y asumir reglas, además de conocer las edades a las que comienzan a aplicar la lógica y adquieren un poder motor que les permite realizar gestos precisos. Teniendo en cuenta estos conceptos y más que se detallan en el anexo, además de los conceptos a transmitir a los niños, podemos definir un rango de edades adecuado como usuario objetivo de este proyecto.

2.2. INTERACCIÓN TANGIBLE

La interacción tangible es una forma de interacción usuario-ordenador en la que lo digital se une con lo físico a través de objetos cotidianos.

Ishii & Ullmer definieron en 1997 las interfaces tangible como “sistemas interactivos en los que objetos convencionales, son al mismo tiempo, control y representación física de la información digital”.

En las interfaces tangibles la interacción entre el usuario y el ordenador es física por lo que la brecha entre el mundo real y el software es mucho menor. En el diseño de estas interfaces se busca el aprovechamiento de las propiedades físicas de los objetos para albergar, mostrar y permitir el control de la información digital ya que estas propiedades de los objetos dan pistas al usuario de como deben ser utilizados.

Un ejemplo de esta interacción son las tabletops, que son mesas convencionales en las que se proyecta imagen de ordenador sobre la superficie y el usuario interactúa con sus manos u otros objetos.

En el *Anexo C - Ejemplos de interacción tangible* se recogen algunos ejemplos de productos de interacción tangible existentes en el mercado. En estos ejemplos se observa como la interacción tangible ofrece una forma de interactuar con una aplicación informática mucho más natural e intuitiva que el modo convencional (ratón y teclado). Esto hace que este modo de interacción sea adecuado para su uso con niños en aplicaciones educativas.

En definitiva, todos los ejemplos del anexo muestran una mayor proximidad entre lo físico y lo digital. En estos casos el atractivo natural de la manipulación física de objetos convencionales acerca a los niños al mundo digital, no sólo suponiendo una fuente de ocio y entretenimiento sino que además se convierten en un recurso educativo.

Estas recientes tecnologías de la interacción tangible descubren en los niños un amplio abanico de nuevos conocimientos y experiencias mostrándoles opciones hasta entonces posiblemente desconocidas para ellos.

Cabe decir, que en la mayoría de los ejemplos propuestos se usa un dispositivo tabletop por ser las mesas un espacio natural de juego con objetos manipulables y por las posibilidades que ofrece al mostrar información digital en el mismo espacio en el que se están manipulando los objetos físicos, los cuales permiten el control, manejo e interacción de la información digital.

Estas características intrínsecas a estos dispositivos de interacción tangible son las ideales para conseguir los objetivos que se proponen en este proyecto por lo que es en estas propiedades en las que se basa el diseño de los juegos.

2.3. TECNOLOGÍAS PARA LA ENSEÑANZA DEL PENSAMIENTO COMPUTACIONAL EN NIÑOS

El pensamiento computacional se puede definir como una herramienta del conocimiento que permite razonar, argumentar y explicar las situaciones y objetos que nos rodean, y paso a paso siguiendo un proceso deductivo y analítico alcanzar las soluciones a los problemas. En el *Anexo D - Pensamiento lógico computacional en niños* se encuentra una definición más extensa de este concepto y los beneficios que su aprendizaje aporta a los niños.

Además, para entender mejor la situación actual de este concepto en el ámbito educativo se muestra en el mismo anexo algunos ejemplos de proyectos llevados a cabo en diferentes países con el fin de inculcar este concepto en los niños. Este breve estudio nos permite ver el creciente interés en el ámbito escolar por la enseñanza del pensamiento lógico en niños, y en consecuencia el interés en seguir trabajando en este ámbito creando nuevos métodos y tecnologías que faciliten la enseñanza de este razonamiento.

La idea de la enseñanza de programación en niños no tuvo cabida hasta que Seymour Papert [30] se planteó que ésto podía conllevar beneficios para éstos. Seymour creó Logo [31] en 1967, el primer lenguaje de programación pensado para la didáctica de programación en niños. Se trata de un lenguaje de alto nivel, funcional, estructurado y muy fácil de aprender, por lo que se utiliza mucho en el aprendizaje en niños y jóvenes.

Con la creación de Logo, el matemático consiguió hacer accesible a los niños un aprendizaje en programación que hasta ese momento nadie se planteaba. Seymour Papert planteó un lenguaje de programación cuya sintaxis fuera apropiada a las características de los niños. Es un lenguaje basado en la escritura con instrucciones fáciles de aprender caracterizado por una sintaxis simple, resultados visuales atractivos y curvas de aprendizaje muy rápidas (Ver Imagen 5).

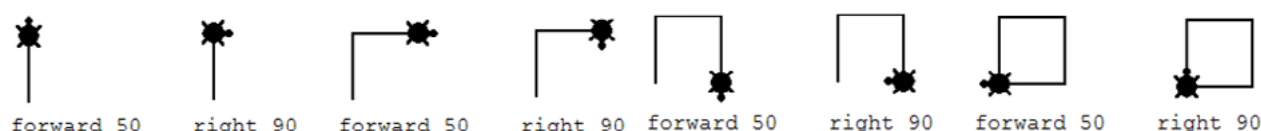


Imagen 5 - Lenguaje de programación Logo. Secuencia de instrucciones para dibujar un cuadrado

Una de las características más destacadas y empleadas de Logo es la capacidad de producir gráficos tortuga (Ver Imagen 6), término usado para referirse a la programación de gráficos vectoriales. El resultado de cada instrucción se ve plasmado en el recorrido efectuado por un robot (tortuga virtual).

A partir de Logo surgen nuevas propuestas y proyectos con sintaxis diferentes para enseñar razonamiento computacional a niños. En este contexto surge en 2002 Scratch [32] (Ver Imagen 7), un nuevo lenguaje de programación visual diseñado por Mitchel Resnick [33] [34] y desarrollado por el MIT Media Lab para niños de entre 8 y 16 años. Con este software educativo libre los usuarios crean fácilmente animaciones y juegos sin necesidad de aprender un lenguaje de programación complejo y evitando los errores de sintaxis. Este len-

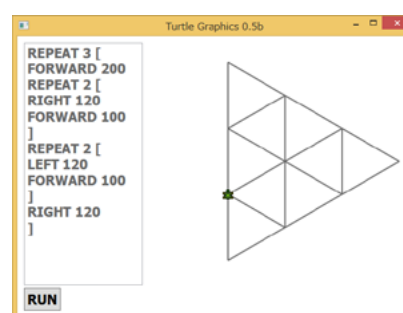


Imagen 6 - Gráficos tortuga, Logo

guaje de programación, al igual que Logo, trata de facilitar el aprendizaje de pensamiento computacional y programación abordando el tema sintaxis, pero de una manera diferente a Logo. Mientras que la sintaxis de logo, es puramente textual (el código debe ser tecleado correctamente mediante un teclado de ordenador), en Scratch los usuarios deben agrupar bloques de diferentes colores y formas para realizar sus propias animaciones y juegos. Estos bloques se agrupan como si fueran puzzles y los colores y formas ayudan a los usuarios a identificar sus funciones, de esta forma se minimizan los errores y esfuerzo de aprendizaje de la sintaxis de un lenguaje de programación.



Imagen 7 - Scratch. Lenguaje de programación por bloques.

Estos dos lenguajes de programación, Logo y Scratch, son muy influyentes y ambos se centran en facilitar el aprendizaje de la programación abordando el tema de la sintaxis. Sin embargo no abarcan el tema de la interacción, siendo ésta convencional (teclado y ratón) en ambos lenguajes. Por otro lado surgen nuevas propuestas basadas en formas de interacción no convencionales, la interacción física. Algunas de estas propuestas llevan Logo a una forma física logrando que la interacción sea tangible y resolviendo así los dos parámetros que dificultan el aprendizaje del pensamiento lógico.

A partir de Scratch surgen nuevas propuestas (juegos y aplicaciones) basadas en su filosofía. De entre todas las propuestas encontradas se han seleccionado y tenido en cuenta aquellas que responden a unos criterios selectivos que son: la necesidad de que enseñen los conceptos de programación, que sean adecuadas y/o específicas para los niños y que sigan la filosofía de Scratch.

En la siguiente tabla se categorizan estas propuestas en base a cuatro aspectos fundamentales que son la edad de los usuarios, el tipo de interacción empleada, la motivación y los conceptos de programación que abarcan. Además ya que estas propuestas surgen a partir de Scratch se incluye éste en la primera fila de la tabla para poder compararlas.

Tabla 4 - Propuestas basadas en Scratch que enseñan programación a niños

		Interacción			Motivación		Conceptos enseñados					
	Edad	Convencional (teclado y ratón)	Apps en dispositivos táctiles	Tangible	Retos	Programación libre		Lenguaje secuencial	Procedimientos	Bucles	Condicionales	Booleanos
						Ver resultados en robot	Ver resultados en app (animaciones)					
Scratch	5+ y 8-16	X					X	X	X	X	X	X
Strawbies [35]	5+			X			X	X	X			
Osmo coding [36]	5-12			X			X	X		X	X	
Light-bot [37]	4-8 y 9+	X	X		X			X	X	X		
Cargo-bot [38]			X		X			X	X	X		
Kodable [39]	5+		X		X			X	X	X	X	
Robot Kibo [40]	4-7			X		X		X				
Cubetto [41]	3+			X		X		X				

Dash y Dot [42]	5-12		X			X		X	X	X		
Zowi [43]	8+		X			X		X	X	X	X	X
Puzzlets [44]	6+			X	X		X	X				

Estas propuestas se encuentran explicadas brevemente en el *Anexo E - Propuestas basadas en Scratch*.

Con respecto al modo de interacción se observa que son pocas las propuestas que continúan apostando por la interacción convencional (teclado y ratón). La gran mayoría se divide entre los que apuestan por el desarrollo de apps para dispositivos táctiles y los que se decantan por la interacción física.

Por otro lado, en el modo de motivar a los niños encontramos dos vertientes muy equilibradas en cuanto a número de propuestas. Las que optan por la superación de retos y las que se decantan por la programación libre a decisión del usuario donde la motivación es la emoción de ver los resultados en un robot físico o mediante animaciones de algún personaje en app en dispositivos táctiles.

Por último los conceptos que se enseñan en cada propuesta son muy variados. Todas enseñan lenguaje secuencial, pero sólo dos tercios enseñan otros conceptos como bucles y procedimientos. Y únicamente un tercio enseña conceptos avanzados como expresiones condicionales o booleanos.

Gran parte de las nuevas iniciativas basadas en Scratch intentan llevarlo al mundo físico. Los ejemplos más claros de esta intención son Strawbies (Ver Imagen 8) y Osmo coding (Ver Imagen 9), en los cuales se programa de igual forma que en Scratch, mientras se alejan de la interacción convencional dejando el ratón y el teclado y sustituyendo los bloques virtuales por bloques físicos que manejar con las manos. El resultado de sus creaciones, al igual que en Scratch, lo ven a través de animaciones en dispositivos táctiles como tablets.



Imagen 8 - Strawbies



Imagen 9 - Osmo Coding

Paralelamente a estas propuestas basadas en Scratch continúan apareciendo nuevas ideas basadas en Logo, como las que se muestran en la siguiente tabla. De entre todas las propuestas encontradas se han seleccionado y tenido en cuenta aquellas que responden a unos criterios selectivos que son: la necesidad de que enseñen los conceptos de progra-

mación, que sean adecuadas y/o específicas para los niños y que sigan la filosofía de Logo.

Tabla 5 - Propuestas basadas en Logo que enseñan programación a niños

		Interacción			Motivación			Conceptos enseñados				
	Edad	Convencional (teclado y ratón)	Apps en dispositivos táctiles	Tangible	Retos	Programación libre		Lenguaje secuencial	Procedimientos	Bucles	Condicionales	Booleanos
						Ver resultados en robot	Ver resultados en app (animaciones)					
Logo	-	X					X	X		X		
Bee-bot [45]	3+			X		X		X				
Blue-bot [46]	5+		X	X		X		X				
Pro-bot [47]	6+			X		X		X	X	X	X	X
TurTan [48]	-			X			X	X				

Estas propuestas se encuentran explicadas brevemente en el *Anexo F - Propuestas basadas en Logo*.

Como se observa en la tabla, las proposiciones más similares a Logo descartan la opción de motivar a los niños mediante retos con los que superarse. Aunque es curioso que los profesores que emplean estos juegos (Bee-bot, Blue-bot y Pro-bot) en el aula suelen plantear retos grupales a sus alumnos.

Con respecto al modo de interacción se observa que todas las propuestas apuestan por la interacción física. El ejemplo más representativo de la idea de llevar Logo a la interacción tangible es TurTan.

TurTan (Ver Imagen 10) ha sido desarrollado por Carles F. Julià, Daniel Gallardo y Sergi Jordà en la Universidad Pompeu Fabra de Barcelona y es un lenguaje de programación tangible orientado al aprendizaje de conceptos básicos de programación inspirado en Logo. La interacción con TurTan es tangible, convirtiendo las instrucciones del lenguaje en piezas físicas, y sus manipulaciones en el tabletop, en configuraciones de los parámetros del lenguaje.

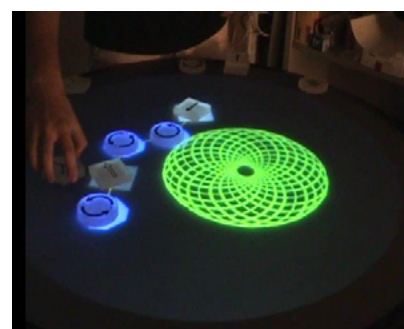


Imagen 10 - TurTan

Debido a la importancia que tiene TurTan en el diseño de las actividades de este proyecto se detalla más información de TurTan en el *Anexo F - Propuestas basadas en Logo*, apartado *F.1. TurTan*, como su sintaxis y modo de interacción.

Como hemos podido observar han surgido nuevas propuestas de formas de interacción que van en la línea de la manipulación física y los juguetes. Cabe destacar que ninguna va hacia otras líneas como la realidad virtual o aumentada, sino que todas las nuevas propuestas se decantan por la interacción tangible para hacer accesible el aprendizaje del pensamiento computacional y la programación.

Por último, es importante analizar el tema de la colaboración en juegos y aplicaciones que

enseñan conceptos de programación. Debemos distinguir entre los juegos que son colaborativos que implican la necesidad de que más de un niño juegue a la vez y los juegos que permiten la participación de más de un niño, pero en los que un solo niño puede jugar, siendo solo colaborativos los primeros.

Dicho esto solo se han encontrado dos juegos colaborativos que son Robot Turtles [49] (Ver Imagen 11) y Code Monkey Island [50] (Ver Imagen 12). Ambos son juegos no tecnológicos y de mesa, lo cual facilita la colaboración. Sin embargo no permiten a los usuarios participar libremente y cooperar para alcanzar la meta del juego. Esto es debido a que los juegos de mesa se rigen por unas normas a seguir, entre las que destaca la participación por turnos, lo que hace que la interacción sea lenta y los niños pierdan el interés por el juego.



Imagen 11 - Robot Turtles



Imagen 12 - Code Monkey Island

2.4. CONCLUSIONES FASE ANÁLISIS

En esta fase de análisis se ha estudiado la evolución de los niños a lo largo de su crecimiento tanto en el ámbito motor como en el intelectual con el fin de conocer a los usuarios potenciales y definir un rango de edad adecuado para los juegos.

En función de este análisis se ha decidido que las edades más apropiadas sean entre los 7 y 10 años. Se ha elegido esta edad porque es el momento en el que empiezan a aplicar la lógica y el razonamiento para entender el mundo que los rodea, lo que facilita el aprendizaje de conceptos de programación más avanzados como los bucles y procedimientos. Además, a esta edad disminuye su egocentrismo, fundamental para realizar juegos en grupos y cooperar correctamente.

Por otro lado se ha estudiado la interacción tangible y los productos existentes que practican este tipo de interacción con el fin de entender las posibilidades que estos nuevos modos de interacción ofrecen en los juegos.

Además existen multitud de proyectos en los que se aplica el pensamiento computacional y la programación como herramienta educativa. Se ha observado que no es hasta los 11 o 12 años cuando los niños comienzan a aplicar los bucles y procedimientos en sus programas, esto lo hacen de la mano de Scratch. Se considera que los niños pueden aprender estos conocimientos a una edad más temprana, aunque Scratch es muy complejo para empezar desde cero. Por ello se quieren diseñar juegos para el rango de edad de entre 7 a 10 años que faciliten el aprendizaje de estos conocimientos.

Se quiere realizar el diseño de juegos que permitan un aprendizaje progresivo abarcando los siguientes conceptos de programación:

- Programación secuencial
- Bucles
- Procedimientos

Por lo tanto nos enfrentamos a un reto principal, que es cómo hacer que entiendan y apliquen los bucles y procedimientos. No parece tan difícil el que lo entiendan, pero si el que lo apliquen y sean capaces de visualizar cuando es necesario, o posible, aplicarlos.

Tras esta fase de análisis y viendo que los dos lenguajes de programación más influyentes como recurso educativo en niños son Logo y Scratch se toman éstos como referencia para el diseño y la creación de juegos por su influencia y su demostrada efectividad en el ámbito educativo.

FASE DE CREACIÓN Y DISEÑO

3

3.1. IDEACIÓN

Teniendo en cuenta el desarrollo social y la evolución de la inteligencia de los niños en la fase previa se ha restringido el rango de edad de los usuarios a entre 7 a 10 años. Además, debemos tener presentes los objetivos marcados y las conclusiones obtenidas en la fase de análisis.

Así, en esta fase nos centraremos en la generación de ideas, la selección y la definición de los conceptos finales. Partimos de un objetivo concreto que es conseguir que los niños aprendan conceptos de programación en un entorno colaborativo e interactivo mientras juegan. A partir de esto y teniendo en cuenta el resto de objetivos se realizan una serie de técnicas creativas con el fin de alcanzar una lista de ideas amplia y variada.

Comenzamos realizando un panel de influencias del usuario objetivo que consiste en recopilar imágenes que representen y definan al usuario y su entorno con el fin de conocerlo mejor y adecuarnos a sus intereses y gustos. Este panel de influencias se encuentra en el *Anexo G - Técnicas creativas aplicadas y conceptos*.

Posteriormente se realizan una serie de mapas mentales con el fin de abrir la mente y facilitar la creación de ideas. Cada mapa mental se ha realizado partiendo de una idea concreta y diferente. Las ideas base son “niños” y “enseñanza” y están en el *Anexo G - Técnicas creativas aplicadas y conceptos*.

También se ha realizado un brainstorming del que se ha obtenido la siguiente lista de ideas que se muestra en el *Anexo G - Técnicas creativas aplicadas y conceptos* junto a las técnicas aplicadas anteriormente.

A partir de las técnicas creativas anteriores se han obtenido conceptos más concretos y definidos que se definen a en el apartado *G.4. Conceptos del Anexo G - Técnicas creativas aplicadas y conceptos*. Estos conceptos han sido el punto de partida para definir los juegos finales a desarrollar, los cuales se describen en los siguientes apartados.

3.2. DEFINICIÓN DE LOS JUEGOS

Tras este proceso de ideación y concretización se obtienen 5 conceptos, que se pueden ver en el *Anexo G - Técnicas creativas aplicadas y conceptos*, y de los cuales de parte para diseñar los dos juegos que aquí se definen.

3.2.1. JUEGO 1: Juego para el espacio JUGUEMOS

SINOPSIS

Los jugadores son astronautas que deben superar unas pruebas para convertirse en tripulantes de naves del espacio exterior. Las proyecciones de las paredes se emplean como “pantalla” compartida donde se muestra un reto común a superar entre todos. Cada niño controla una nave distinta y debe recoger las estrellas de su color para pasar al siguiente nivel. Para manejar las naves cada niño dispone de una mesa donde éstos deben construir sus programas con fichas físicas. Para superar cada uno de los niveles todos los jugadores deben superar los retos propuestos, por lo que pueden ayudarse unos a otros para aprender juntos y superar estos retos.

SINTAXIS DEL LENGUAJE DE PROGRAMACIÓN

Para controlar la nave espacial los niños disponen de cinco tipos de fichas con las que manejar las naves a través del espacio.

1. **Flecha de dirección.** Esta ficha tiene 4 posiciones (avanzar, retroceder, izquierda y derecha) y permite avanzar al usuario por el tablero del juego para lograr el reto que se propone. (Ver Imagen 13)



Imagen 13 - Boceto cuatro posiciones de la ficha “Flecha de dirección”

2. **Bucle.** La ficha de bucle se emplea detrás de las instrucciones a repetir (Ver Imagen 14). Para definir qué instrucciones debe abarcar se debe dejar una separación (de una casilla) entre éstas y las anteriores, en el caso de haberlas. Cuando el niño ponga una ficha de bucle la mesa indica que instrucciones abarca con un rectángulo que las encuadre y resalte y muestra un menú circular alrededor de la ficha de bucle que los niños deben rotar para seleccionar el número de repeticiones que desean ejecutar. (Ver Imagen 15 y 16)

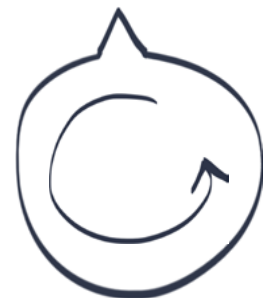


Imagen 14 - Boceto ficha “Bucle”

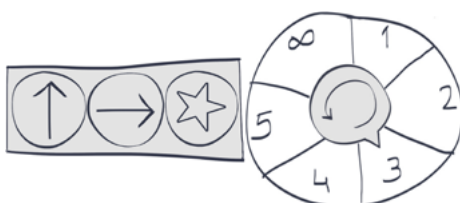


Imagen 15 - Boceto de un bucle de tres fichas

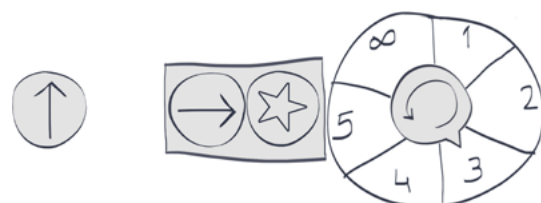


Imagen 16 - Boceto de una ficha seguida de un bucle de dos fichas

- 3. Procedimiento azul.** Esta instrucción se emplea para ejecutar las instrucciones definidas en la línea del procedimiento azul. Para que el programa ejecute el procedimiento azul definido hay que colocar una ficha de procedimiento azul (o tantas como se deseen ejecutar) en la línea en la que estemos haciendo el programa a ejecutar. Además, para definir la línea de procedimiento azul se debe colocar una ficha de éste en la línea al principio o al final, y la mesa resaltará esa línea de color azul. (Ver Imagen 17)



Imagen 17 - Boceto ficha "Procedimiento azul"

- 4. Procedimiento rosa.** Esta instrucción se emplea del mismo modo que la anterior, únicamente cambia el color para diferenciarlas y así poder definir dos procedimientos diferentes. (Ver Imagen 18)



Imagen 18 - Boceto ficha "Procedimiento rosa"

- 5. Recoger estrella.** Los usuarios deben usar esta ficha para recoger todas las estrellas de su color. Cada vez que el programa ejecuta esta ficha se recoge la estrella que este en la misma casilla en la que se encuentre la nave en ese momento. Si la nave se encuentra en una casilla en la que no hay estrella y se ejecuta esta ficha no ocurre nada y se continua ejecutando el resto del programa. (Ver Imagen 19)



Imagen 19 - Boceto ficha "Recoger estrella"

ZONAS DE INTERACCIÓN

Las mesas tienen tres líneas donde el niño podrá colocar las fichas a su antojo. Estas líneas son para los procedimientos azul y rosa y para el programa principal y es el niño quien decide como usarlas. (Ver Imagen 20)

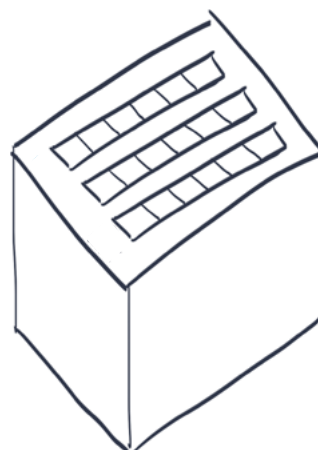
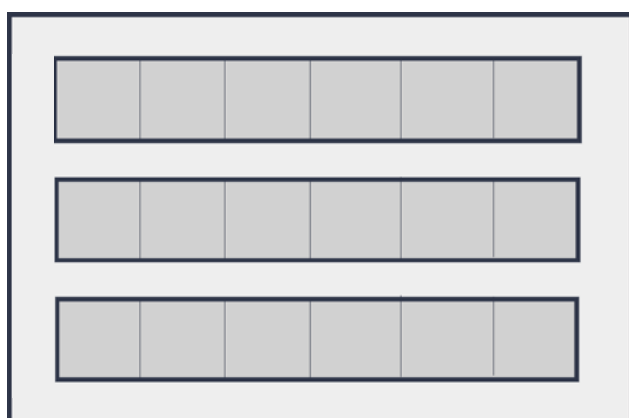


Imagen 20 - Boceto superficie de las mesas NIKVision y mesa NIKVision

Todo el programa principal debe estar en una sola línea. Si no cabe se deben emplear bucles o procedimientos para ahorrar espacio en la línea del programa principal. La descripción de los procedimientos irá en otra línea y hay espacio para dos procedimientos diferentes aun-

que no sea necesario usarlos.

En las paredes se proyecta el reto a resolver en el espacio de juego por donde deben desplazarse para recoger las estrellas. Este espacio está señalado y delimitado por una cuadrícula, como se muestra en el boceto (Ver Imagen 21). El juego se desarrolla en la pared principal, donde la cuadrícula ocupa prácticamente toda la pared, mientras que en las paredes laterales se muestran imágenes decorativas que no influyen en el juego, simplemente son estéticas para conseguir un efecto más envolvente.

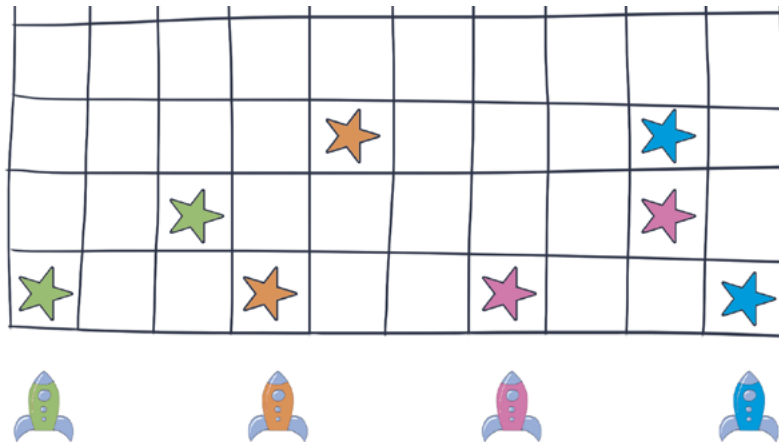


Imagen 21 - Boceto espacio o "tablero" de juego. Cuadrícula

El feedback dentro del espacio JUGUEMOS debe ser continuo y fluido para que los usuarios estén informados del estado del juego. Para ello los niños deben ver las respuestas y consecuencias de sus acciones, sean buenas, malas o neutras.

- Si las acciones son buenas, es decir, se realizan correctamente se muestra a los niños una respuesta positiva en el entorno.
- Si las acciones son neutras, es decir, no afectan al transcurso del juego no se muestra ninguna respuesta.
- Si las acciones son malas, es decir, afectan negativamente al desarrollo del juego se les da a los niños una nueva oportunidad. Nunca se deben dar respuestas negativas a los usuarios.

SECUENCIA DEL JUEGO

Además, es un juego por niveles para cuatro jugadores. Los niveles están ordenados por dificultad ascendente creando cuatro unidades didácticas en las que se enseñan los conceptos definidos, que son lenguaje direccional, procedimientos, bucles y por último todos los conceptos juntos.

Los niveles y vídeos del juego se distribuyen de la siguiente forma:

- El juego está a punto de comenzar
- Vídeo de bienvenida e instrucciones principales (flecha y estrella)
- Juego 1
- Juego 2
- Vídeo instrucciones procedimiento azul
- Juego 3
- Juego 4
- Vídeo instrucciones procedimiento rosa
- Juego 5
- Juego 6

- Video instrucciones bucle
- Juego 7
- Juego 8
- Fin del juego

Al comienzo de cada nivel se les explican los nuevos conceptos de manera visual con vídeos. Para esta tarea se necesita un personaje que protagonice los vídeos explicativos. Dado que finalmente el entorno el juego es el espacio se plantean dos opciones: un extraterrestre o un astronauta, decantándose finalmente por la primera opción. A continuación se muestran algunos bocetos de extraterrestres (Ver Imagen 22).



Imagen 22 - Bocetos Spowi

Este personaje es el protagonista de los vídeos y se encarga de dirigir el juego y acompañar a los niños a lo largo de éste. Este personaje debe tener nombre y para la elección de éste se trata de no repetir ningún nombre de algún personaje ya existente y se buscan nombres amigables, fáciles de recordar y con una sonoridad agradable y divertida. Finalmente, tras una pequeña investigación y una pequeña generación de ideas el nombre del extraterrestre es **Spowi**.

NOMBRE DEL JUEGO

Para la elección del nombre del juego se busca que el nombre tenga una buena sonoridad y que tenga relación con el juego. Por ello se realiza una lista de ideas para el nombre del juego, tanto en castellano como en inglés (debido al atractivo sonoro de las palabras en inglés).

Space runner
Space team
Space squad
Space mission

Space task
Space guardians
Guardians of space
Space explorers

Galactic travellers
Space travellers
Space patrol
Cosmonauta

Astronauta	Spacemanbot	Galactic loop
Tripulantes espaciales	Space programmers	Spowi's galactic trip
Disfruta y aprende con Spowi	Loop space	The galactic tale of spowi
Enjoy and learn with Spowi	Loop galaxy	The rise of spowi
Code Spowi	Star's loop	Spowi's loop
Space Code	Star loop	Captain spowi: a loop story
Spacial code	Stars' loop	How to be a programmer with Spowi
Código espacial	Bucle de estrellas	Code for children with spowi
Galaxy code	Bucle estelar	Guardians of stars
Galactic code	Space loop	Star's guardians
Código galáctico	Bucle espacial	Falling star
Galactic programmers	Galactic loop	
Spaceman	Spatial loop	

A partir de esta lista de ideas se seleccionan algunos nombres, subrayados en amarillo, y se elige entre éstos buscando un nombre original y que defina el juego o algún aspecto de éste. Finalmente el nombre elegido es **Star Loop**. La elección de este nombre se debe al juego de palabras de star (estrella) y loop (bucle), creando Star Loop que significa “bucle estelar”. Así este nombre hace referencia al ámbito del juego, el espacio, y a lo que deben realizar los niños que es recoger estrellas, además del hecho de que se aprende el concepto de los bucles. Además se ha elegido este nombre en inglés por su sonoridad y su estética debido a que son dos palabras muy cortas y ambas de cuatro letras, lo cual favorece la creación de un logotipo para el juego.

Una vez elaborado el primer diseño de concepto de Star Loop se desarrollan los recursos multimedia y un documento explicativo del juego necesario para la realización y codificación del prototipo con el fin de que éste pueda ser probado, evaluado y refinado en el proceso iterativo que se detalla en la siguiente fase del proyecto. Este documento explicativo se encuentra en el *Anexo H - Star Loop*, apartado *H.2. Documento explicativo inicial*.

3.2.2. JUEGO 2: Ampliación de TurTan para NIKVision redonda

Como se ha explicado anteriormente es un juego ya existente desarrollado por Carles F. Julià, Daniel Gallardo y Sergi Jordà en la Universidad Pompeu Fabra de Barcelona y es un lenguaje de programación tangible orientado al aprendizaje de conceptos básicos de programación inspirado en Logo.

Resulta interesante la idea de aprender conceptos básicos de programación de la mano de gráficos tan vistosos y atractivos como los que resultan al mezclar unas pocas instrucciones. Además, al coincidir la realización de este proyecto de final de grado con la creación de una nueva mesa NIKVision con forma redonda resulta adecuado y oportuno el desarrollo de alguna aplicación para ésta. La forma redonda de la mesa al no tener ejes de simetría predominantes facilita el diseño de aplicaciones donde no se priorice ninguna dirección, es decir, donde todos los usuarios puedan interactuar con la mesa desde cualquier ángulo y posición en igualdad de condiciones. Estas características permiten el diseño y desarrollo de aplicaciones para niños de carácter cooperativo donde éstos puedan colaborar y aprender mano a mano.

Se dan tres circunstancias importantes, que son la disponibilidad de una mesa NIKVision redonda y la existencia de TurTan y adecuación al tema del presente proyecto. Por ello, al juntarse estos dos factores clave se ha tomado la decisión de realizar una ampliación de este juego y una posterior evaluación con usuarios reales, dado que el potencial de TurTan aun no ha sido testado con niños.

Se recuerda que en el *Anexo F - Propuestas basadas en Logo*, apartado *F.1. TurTan* se encuentra la sintaxis original de TurTan. A partir de este juego se buscan ideas para ampliarlo y las fichas propuestas para añadir a la sintaxis son las siguientes:

1. **Cambiar color.** Cambia el color de la línea para los siguientes movimientos de la tortuga. (Esta ficha ya existe en la sintaxis original de TurTan, pero no queda definido su efecto, por lo que la incluimos en la sintaxis propuesta para definir su efecto a nuestro gusto).
2. **Cambiar grosor.** Cambia el grosor de la línea para los siguientes movimientos de la tortuga.
3. **Espejo radial (efecto kaleidoscopio).** Con esta ficha se quiere conseguir un efecto de espejo que al girar la ficha aumente el número de espejos creando un efecto de kaleidoscopio. Los espejos se crean con respecto al origen de la tortuga y no con respecto de la posición actual.
4. **Polígono regular.** Esta ficha dibuja polígonos regulares, empezando por el triángulo equilátero y aumentando el número de lados del polígono conforme se gira la ficha.
5. **Zoom.** Sirve para aumentar o disminuir los gráficos creados para verlos bien, ya que a veces los gráficos sobresalen de la mesa y no se ve la tortuga o se hacen tan pequeños que tampoco se ve la tortuga. Esta ficha no se incluye en la secuencia de acciones del programa, es individual, por lo que se puede poner en cualquier parte de la mesa sin alterar el programa creado. El efecto de esta ficha permanece tras retirarla de la mesa.

6. **Animar.** Crea animaciones con los gráficos dibujados. Esta ficha no se incluye en la secuencia de acciones del programa, es individual, pero afecta a la ficha del programa más cercana modificando su valor en el tiempo, creando animaciones visuales e inesperadas.
7. **Guardar.** Esta ficha almacena la secuencia de fichas colocadas a su izquierda. Una vez guardado se pueden recoger todas las fichas y empezar de nuevo empleando esta ficha como una instrucción más, pudiendo modificar su escala, color, grosor, etc.

Una vez definidas las nuevas instrucciones del juego TurTan se procede al desarrollo de los diseños gráficos de las fichas, ya que no son necesarios recursos gráficos para el desarrollo informático de estas instrucciones. Destacar que esta implementación informática es llevada a cabo por el director de este proyecto, y una vez implementado el prototipo se prueban estas nuevas instrucciones con el fin de asegurar que el resultado visual es el deseado y ajustar la velocidad de giro de las fichas físicas con la velocidad de modificación de los parámetros hasta conseguir el efecto deseado.

FASE DE DESARROLLO E IMPLEMENTACIÓN

4

4.1. STAR LOOP



DESARROLLO DEL JUEGO STAR LOOP

El desarrollo del juego abarca todo el trabajo necesario hasta la obtención de un prototipo final que poder evaluar con niños. Este trabajo implica la creación de todos los recursos necesarios y la implementación de éstos. Para la realización de éstos recursos se han tenido en cuenta una serie de requisitos técnicos y del propio espacio JUGUEMOS que se detallan en el *Anexo H - Star Loop*, apartado *H.1. Restricciones técnicas e información necesaria para desarrollar los recursos*.

Cabe decir, que ha sido fundamental el trabajo llevado a cabo por la becaria, estudiante del grado de ingeniería informática, la cual ha llevado a cabo la implementación y desarrollo informático del juego. Además la buena comunicación y entendimiento entre el diseñador y el desarrollador, la becaria, es fundamental para llevar a cabo el desarrollo de Star Loop. Para que esta comunicación sea adecuada se debe elaborar un documento explicativo donde se detalle todo lo concerniente al juego, los recursos elaborados y su implementación, de manera que no quede nada sin definir con el fin de facilitar el trabajo del desarrollador.

Una vez desarrollado el prototipo inicial del juego está listo para ser probado y modificado hasta alcanzar el juego final que probar con los niños. Este proceso que comienza tras la primera implementación del juego es un proceso iterativo en el que mediante pruebas realizadas por usuarios expertos (adultos) se buscan fallos con el fin de mejorar el juego para la evaluación final con usuarios reales (niños). Añadir que estas pruebas se realizan por el director del trabajo, la becaria que implementa el juego y el diseñador del juego con el prototipo funcional y en el espacio JUGUEMOS.

En el siguiente esquema queda reflejado el proceso del desarrollo del juego, incluyendo el proceso iterativo llevado a cabo. A continuación se explica y detalla la evolución del juego durante este proceso iterativo y el porqué de esta evolución. La creación de los recursos y modificación de éstos se muestra en el *Anexo H - Star Loop*.

FASE DE DESARROLLO E IMPLEMENTACIÓN

PROTOTIPO
INICIALFASE DE CREACIÓN
Y DISEÑOGENERACIÓN
DE RECURSOS
INICIALES

(Doc. explicativo y recursos)

IMPLEMENTACIÓN

PRUEBA 1

MODIFICACIÓN
DE RECURSOS

(Doc. explicativo y recursos)

IMPLEMENTACIÓN

PRUEBA 2

MODIFICACIÓN
DE RECURSOS

(Doc. explicativo y recursos)

IMPLEMENTACIÓN

PRUEBA 3

MODIFICACIÓN
DE RECURSOS

(Doc. explicativo y recursos)

IMPLEMENTACIÓN

PRUEBA 4

MODIFICACIÓN
DE RECURSOS

(Doc. explicativo y recursos)

IMPLEMENTACIÓN

PRUEBA 5

FASE DE
EVALUACIÓN Y
RESULTADOSTrabajo llevado a
cabo por la becaria
informática

PROTOTIPO INICIAL

El objetivo de este primer prototipo inicial es tener disponible en poco tiempo un prototipo funcional que permita probar el lenguaje de programación creado y evaluar su uso e interacción lo antes posible para detectar problemas o dificultades de uso no detectables en el diseño del concepto.

Para este prototipo inicial únicamente se generan los recursos necesarios para que éste prototipo sea funcional. Para este primer prototipo se realizan 8 niveles progresivos de menor a mayor dificultad que se dividen en grupos de dos niveles para formar cuatro unidades didácticas en las que aprenden un concepto nuevo. Estas unidades son las siguientes:

1. Programación secuencial
2. Un procedimiento
3. Dos procedimientos
4. Bucle

El bucle es un concepto que en la mayoría de productos y tecnologías existentes en el mercado se aprende antes que los procedimientos, pero a pesar de esto se considera de mayor dificultad por el hecho de incluir un menú radial y la necesidad de entender bien el manejo de la ficha para controlar que secuencia de instrucciones incluir. Por este motivo este concepto se ha dejado para el final, pero no se descarta adelantarlo a los procedimientos si se observa que no es tan complicado como parece ser de primeras.

Este prototipo incluye los recursos gráficos básicos para las mesas NIKVision (Ver Imagen 23), las imágenes para cuando se define un procedimiento azul o rosa y el menú radial de los bucles con el rectángulo morado que enmarca la secuencia a repetir (Ver Imagen 24).

Por otro lado, para las proyecciones de las paredes únicamente se crean los recursos gráficos para la pared central que es donde está la cuadrícula del juego. Se crea un fondo para los niveles con la cuadrícula de juego (Ver Imagen 25), donde el desarrollador coloca las naves y estrellas (Ver Imagen 26 y 27) en las posiciones definidas en el documento explicativo. La ejecución de los programas se ordena desde un mando inalámbrico controlado por un adulto. Tras cada ejecución aparece una pantalla de calificación que incluye una imagen negra con una opacidad del 60%, unos mensajes (“¡Reto conseguido!” y “Ohh noo... ¡Vuelve a intentarlo!”) y animaciones individuales para cada usuario que les indican si han superado el reto o no. Cuando los cuatro usuarios superan el reto se pasa de nivel y aparece otra animación anunciando el nivel correspondiente.

Cabe destacar, la creación de unos vídeos explicativos y todos los recursos necesarios para ello, como un personaje principal, Spowi. Estos vídeos no son imprescindibles para



Imagen 23 - Fondo mesa

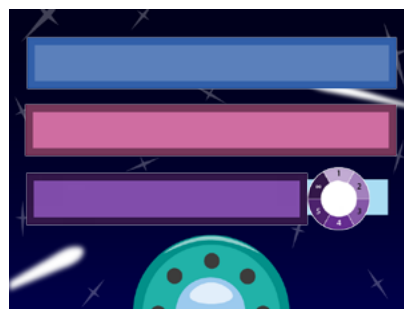


Imagen 24 - Mesa con líneas resaltadas

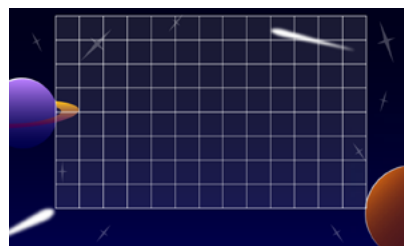


Imagen 25 - Proyección central pared con cuadrícula



Imagen 26 - Estrellas



Imagen 27 - Nave

el prototipo inicial, pero se realizan para comprobar su entendimiento y adelantar trabajo. Se realizan tres vídeos, el primero da la bienvenida al juego y explica las fichas básicas para la primera unidad didáctica (flecha y estrella), el segundo y tercer vídeo explican los procedimientos azul y rosa respectivamente y el último explica el concepto del bucle.

Además, se buscan recursos auditivos en bancos de sonido gratuitos y se seleccionan algunos audios para representar las siguientes circunstancias: recoger estrella, reto individual superado (sonido individual para cada mesa), reto individual no superado (sonido individual para cada mesa), nivel superado, repetir nivel y una música de fondo para el juego.

Por último se elaboran los recursos físicos. Las fichas (88 fichas en total para que cuatro jugadores puedan superar todos los niveles) son recortadas con cúter en cartón pluma debido al excesivo número de fichas necesarias y la facilidad y rapidez de prototipado de este material. A estas fichas se les pega por un lado los diseños realizados para identificarlas correctamente y por el otro, los fiduciales correspondientes en la posición adecuada. Para conocer más detalles sobre su realización o el número de fichas de cada tipo ver en el *Anexo H - Star Loop*, apartado *H.3. Creación de recursos físicos*.

Para facilitar la comunicación con el desarrollador informático del juego y que todo quede bien explicado se realiza un documento explicativo donde se indica todo lo referente a los recursos y su implementación. Este documento se encuentra en el *Anexo H - Star Loop*, apartado *H.2. Documento explicativo prototipo inicial*. Con este prototipo inicial implementado se realiza la primera prueba.

PRUEBA 1

Esta prueba se realiza en el espacio JUGUEMOS, allí nos reunimos el director del trabajo de fin de grado, el desarrollador (becaria) y el diseñador del juego (yo) para probar el prototipo implementado y encontrar los primeros problemas, tanto de la implementación informática, como de diseño del juego y creación de recursos.

En la siguiente tabla se recogen los problemas encontrados y las ideas propuestas para solventarlos.

Tabla 6 - Problemas encontrados en la prueba 1 y cambios realizados tras ésta

Problemas encontrados	Cambios realizados
En el menú radial del bucle los números indican el número de repeticiones (incluyendo la primera ejecución), por lo que no tiene sentido que empiece en el número 1.	Modificar el menú radial para que la repetición más baja sea 2. (2, 3, 4, 5, 6 y infinito) (Ver Imagen 28)
El menú radial del bucle es muy grande y ocupa demasiado espacio, además de que se sale de la línea.	Ajustar el tamaño del menú radial con respecto a la ficha de bucle y las líneas. Fichas de bucle 6 cm. Menú radial 9 cm. Líneas 9 cm. (Ver Imagen 28 y 29)
La detección de las fichas a veces falla.	Dividir las líneas en casillas visualmente y que estas casillas sean más grandes que las fichas, para poder definir rangos de detección sin que se solapen unas con otras. Líneas de 9 cm y fichas de 7 cm. Con casillas de este tamaño caben hasta 7 fichas en cada línea. (Ver Imagen 29 y 30)
Que el uso de las líneas de la mesa estén sin definir y lo decida el usuario parece un problema más para éste, ya que puede dudar de como se usan o porque hay tres iguales.	Asignar a cada línea un uso (procedimiento azul, procedimiento rosa y programa principal). Así evitamos emplear una ficha de procedimiento para definirlo y sólo se emplean para De esta forma no es necesario emplear una ficha de procedimiento para definirlo, únicamente para ejecutar la secuencia creada en la línea de procedimiento correspondiente. (Ver Imagen 30)
No hay feedback con las mesas. No se sabe si una ficha es detectada correctamente o no.	Recurso casilla ocupada. (Ver Imagen 30)

A continuación se muestran los recursos modificados.

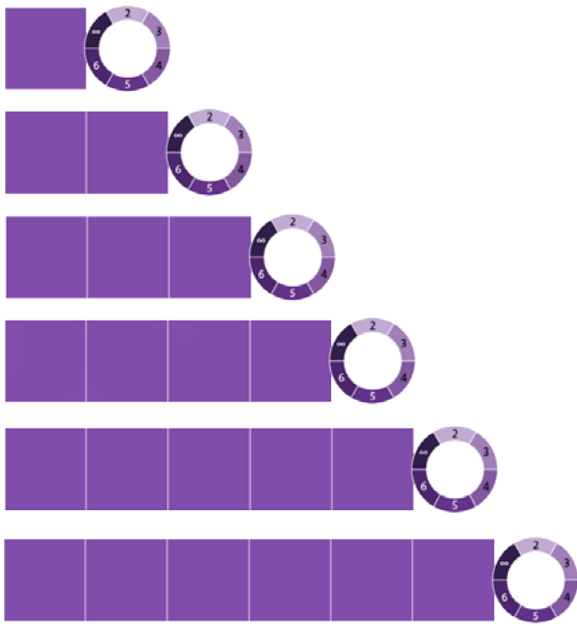


Imagen 29 - Recursos gráficos para enmarcar las secuencias del bucle. La secuencia puede ser desde 1 ficha hasta 6. Con separación de casillas.

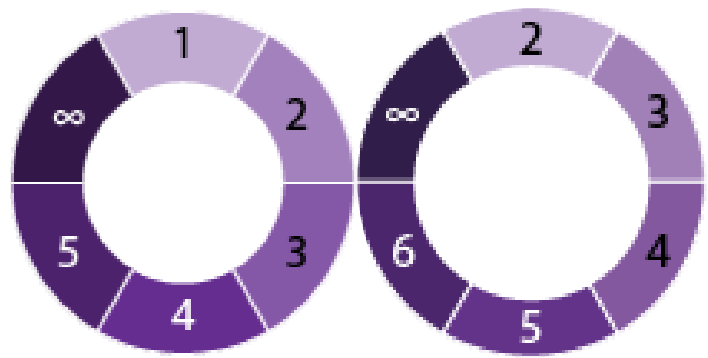


Imagen 28 - Antes y después del cambio del menú radial del bucle



Imagen 30 - Recurso fondo de las mesas NIKVision. Líneas modificadas casillas separadas y con uso definido y a la derecha las casillas ocupadas de cada línea.

PRUEBA 2

Tras las modificaciones necesarias y la implementación por parte de la becaria, nos volvemos a reunir en el espacio JUGUEMOS para examinar de nuevo el prototipo. En esta prueba encontramos los siguientes problemas.

Tabla 7 - Problemas encontrados en la prueba 2 y cambios realizados tras ésta

Problemas encontrados	Cambios realizados
Cuando se ejecutan los programas te pierdes el recorrido que realizan las naves porque va muy rápido	Ralentizar la velocidad de ejecución y mostrar el recorrido visualmente con líneas de colores. (Ver Imagen 31 y 32)
Los niños tienen que esperar a que el resto acabe para ejecutar todos los programas a la vez	Que cada niño tenga una ficha de su nave que sirva para ejecutar su programa cuando él quiera, sin necesidad de que el resto haya acabado. Deben colocar la ficha en la nave de sus mesas para ejecutar sus programas.
Si cada niño ejecuta su programa cuando quiere no puede haber pantalla de calificaciones. Hay que mostrar en la misma pantalla de juego el resultado y sin entorpecer a los otros niños.	Se ejecuta el programa sin paralizar el juego para el resto de jugadores y después se muestra la calificación (también sin paralizar el juego). (Ver Imagen 32) En la posición inicial de la nave: Los mensajes de bien o mal, el número de estrellas (de las dos últimas ejecuciones) y las animaciones de las barras. (Ver Imagen 32 y 34) En la posición final de la nave: La animación de la nave con estrellas o si no se recogen todos los retos ninguna animación. (Ver Imagen 32)
Cuando no recoge una estrella no se muestra de ninguna forma. Necesario feedback con el juego	Debe sonar un sonido de error cuando se ejecuta un “coger estrella” y no hay ninguna estrella en esa posición.
¿Qué pasa si sale del tablero? ¿Puede volver o debe intentarlo de nuevo?	Si sale del tablero debe volver a realizar el programa y suena un sonido de error. Además aparecen las calificaciones para indicar que ha terminado la ejecución
No hay feedback en el menú radial de los bucles y no se sabe si al girar la ficha seleccionas bien el número de repeticiones.	Hacer imágenes para distinguir cuantas repeticiones se seleccionan en el menú radial del bucle. (Ver Imagen 35)
¿Por que instrucción va la ejecución? Me he perdido	Indicar la instrucción ejecutada en cada momento resaltando la casilla de color negro.
No se sabe cuantas estrellas se han recogido y cuantas no. Los usuarios no tienen la suficiente información de la situación.	Hay dos contadores de estrellas. Uno muestra las estrellas recogidas en el intento anterior (este contador no aparece cuando es el primer intento). Y otro contador que se actualice conforme se recogen las estrellas para comparar y estar actualizado de lo que se consigue. Estos contadores son recursos generados directamente por el desarrollador informático y aparecen cuando se ejecuta el programa. (Ver Imagen 32)

A continuación se muestran los recursos modificados.



Imagen 31 - Líneas horizontales y verticales para representar el recorrido de la nave.

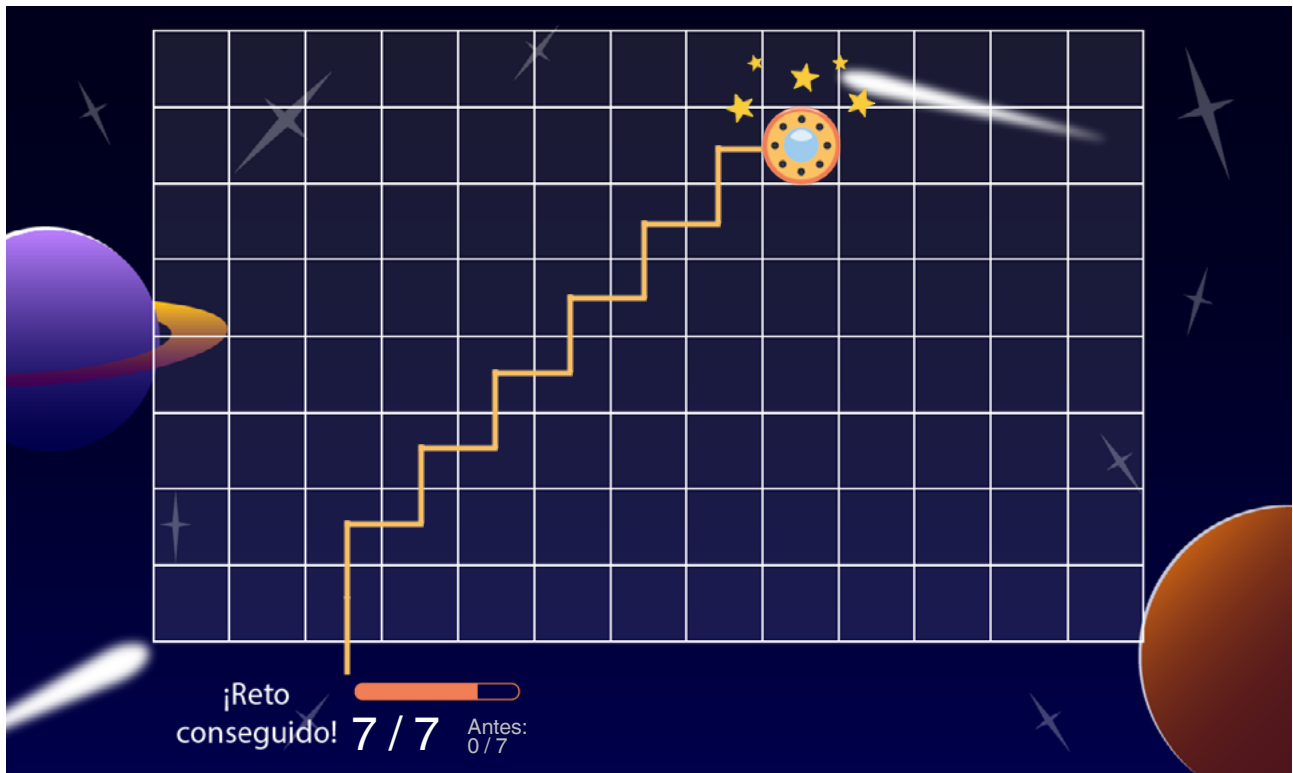


Imagen 32 - Pantalla con recorrido de la nave y calificación.

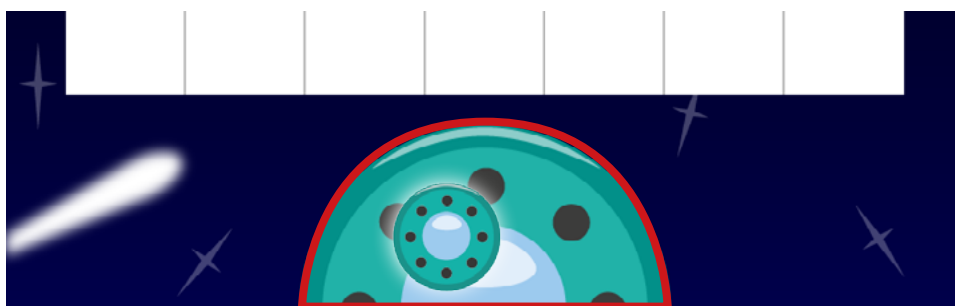


Imagen 33 - Mesa con el área de ejecución resaltada en rojo y una ficha de nave sobre ella.



Imagen 34 - Los dos posibles mensajes de calificación.

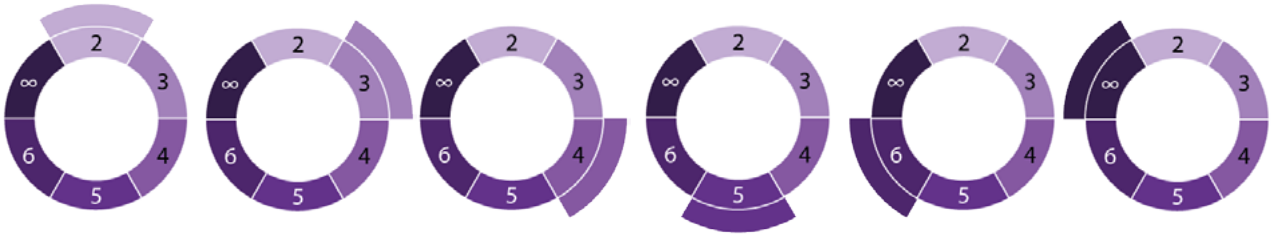


Imagen 35 - Recursos para saber que número de repeticiones se selecciona en el menú radial del bucle.

PRUEBA 3

En la tercera prueba encontramos una serie de problemas de diseño que se detallan en la siguiente tabla, pero también encontramos problemas de software que impiden el transcurso del juego debido a que colapsan la memoria del ordenador. Ésto se debe a que las animaciones exportadas no tienen el tamaño real que presentan en pantalla durante el juego o tienen más fondo del estrictamente necesario. Para solucionar estos problemas hubo que reexportar varias animaciones a su tamaño adecuado, solventando los problemas de carga de recursos gráficos en el ordenador del sistema JUGUEMOS.

Tabla 8 - Problemas encontrados en la prueba 3 y cambios realizados tras ésta

Problemas encontrados	Cambios realizados
Si no hay cuatro jugadores hay que hacer el papel de un cuarto jugador para poder pasar de nivel	Dar la posibilidad de que no sean necesarios cuatro jugadores. Crear una pantalla inicial que pregunte “¿Cuántos jugadores quieren jugar?” y crear una cuenta atrás de 10 segundos para que coloquen la ficha de la nave en la mesa del color correspondiente. (Ver Imagen 36) En las mesas les aparece un mensaje diciendo “Si quieres jugar coloca tu ficha aquí” y cambia cuando detecta la ficha de la nave (Ver Imagen 37 y 38). La cuenta atrás facilita el que no se necesite la interacción de un adulto.
No se aprecia con claridad cuando las naves cogen una estrella.	Crear una animación para cuando se coge una estrella. Además las naves deben pasar por debajo de las estrellas para no ocultarlas en ningún momento. (Ver Imagen 41)
No se distinguen bien las estrellas de las naves cuando están en la misma casilla.	Modificar las estrellas para que se vean bien cuando la nave pasa por debajo. (Ver Imagen 39 y 40)
¿Cómo se sabe que se está ejecutando el programa?	Animación de ejecutando en la mesa (se muestra mientras la ficha de ejecutar esté sobre la nave de la mesa) (Ver Imagen 42 y 43). Mensaje de “Preparados para el lanzamiento” y una cuenta atrás en la posición inicial de la nave para que presten atención a la ejecución del programa. El mensaje de la mesa y la cuenta atrás de la pared desaparecen al finalizar ésta. (Ver Imagen 43, 44 y 45)
La progresión de los niveles de dificultad es muy rápida.	Crear más niveles de adaptación iniciales (juego 1 estrella, juego 2 estrellas, etc.). Se añaden nuevos niveles para que la evolución sea más progresiva. Programación secuencial (5 niveles), Bucle (4 niveles), Procedimiento azul (1 nivel) y Procedimiento rosa (2 niveles). (Ver Imágenes 52 a 63)
Las fichas se mezclan	Hacer un archivador para tener las fichas organizadas

El orden de las unidades didácticas no es el mejor. Se cree que es más sencillo de entender los bucles, por lo que deben ir antes que los procedimientos. Además en los procedimientos se pueden usar los bucles, por lo que tiene sentido que se aprendan los bucles antes.	Cambiar el orden de las unidades didácticas: programación secuencial, bucles, un procedimiento y dos procedimientos
Los vídeos explicativos están desactualizados	Tras este cambio en la secuencia del juego es necesario modificar los vídeos explicativos de Spowi y reorganizar los conceptos, además de incluir la explicación de la ficha de la nave que aún no está.
Las naves azul y verde se diferencian poco en las mesas y paredes	Cambiar los colores de las naves azul y verde (Ver Imagen 46)
Las animaciones ocupan mucho espacio porque están realizadas del tamaño de la proyección central lo cual provoca errores de memoria en ToyVision, ya que se sobrepasa el límite de memoria RAM de los equipos del espacio JUGUEMOS	Recortar las animaciones, deben abarcar únicamente los píxeles imprescindibles, sin dejar fondo vacío alrededor de la animación. (Ver Imágenes 47 a 51)

A continuación se muestran los recursos modificados.

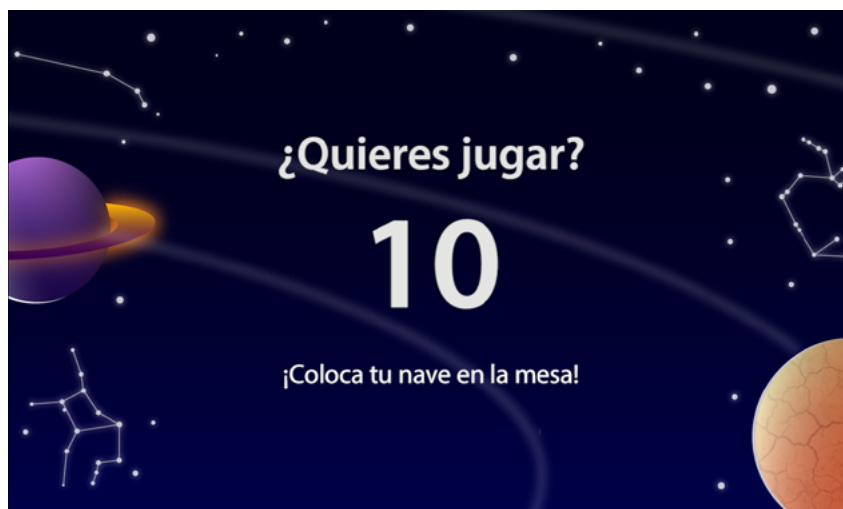


Imagen 36 - Vídeo de cuenta atrás de 10 segundos para seleccionar el número de jugadores



Imagen 37 - Imágenes de las mesas NIKVision mientras se proyecta el vídeo de la cuenta atrás y no se detecta ninguna ficha de nave

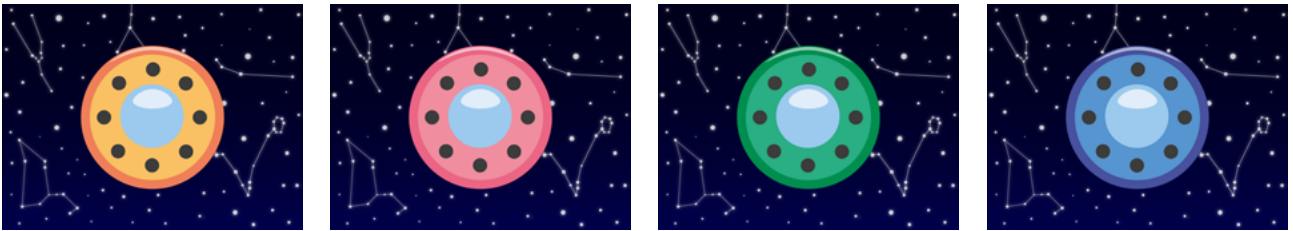


Imagen 38 - Imágenes de las mesas NIKVision mientras se proyecta el vídeo de la cuenta atrás y se detecta una ficha de nave



Imagen 39 - Recursos gráficos de las estrellas antes y después de ser modificados.



Imagen 40 - Estrella sobre nave antes y después de modificar las estrellas

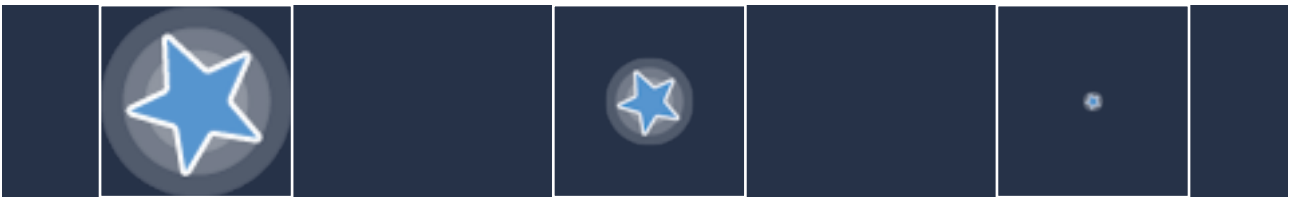


Imagen 41 - Secuencia de tres fotogramas de la animación abducir estrella

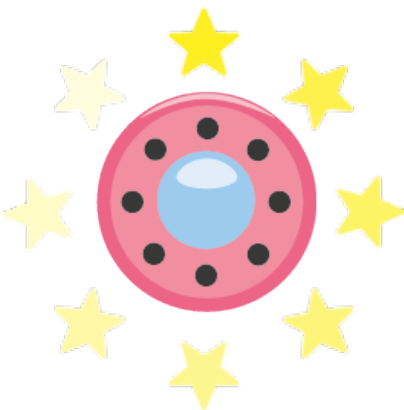


Imagen 42 - Animación de ejecutar. Se muestra sobre la mesa alrededor de la ficha de la nave.

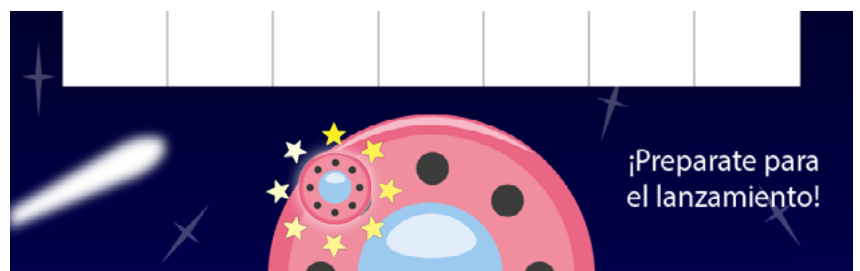


Imagen 43 - Animación de ejecutar y mensaje "¡Preparate para el lanzamiento!" sobre la mesa NIKVision al colocar la ficha de la nave en el área de detección de la mesa



Imagen 44 - Secuencia de la cuenta atrás del lanzamiento que se muestra en la posición inicial de la nave cuando se coloca la ficha de la nave en el área de detección de la mesa

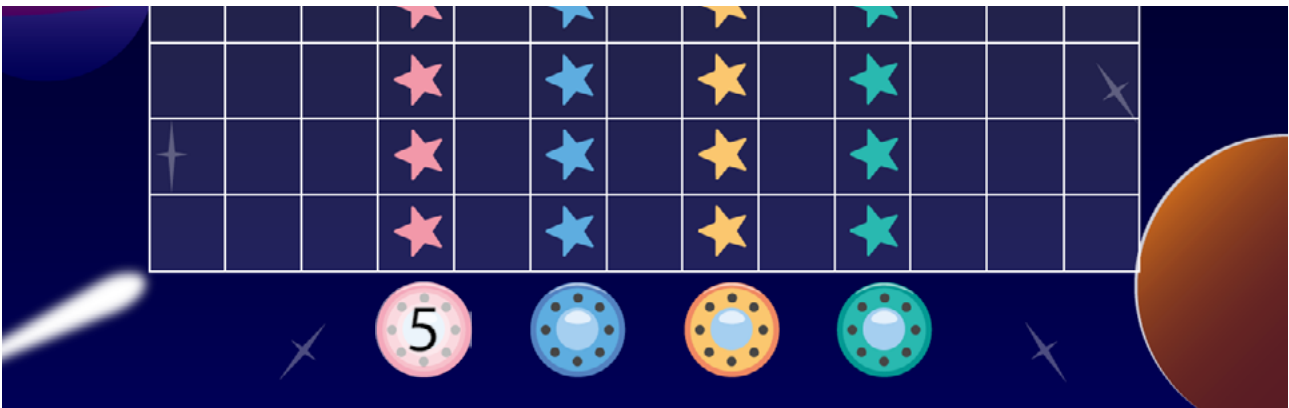


Imagen 45 - Secuencia de la cuenta atrás del lanzamiento en su posición sobre la proyección de la pared

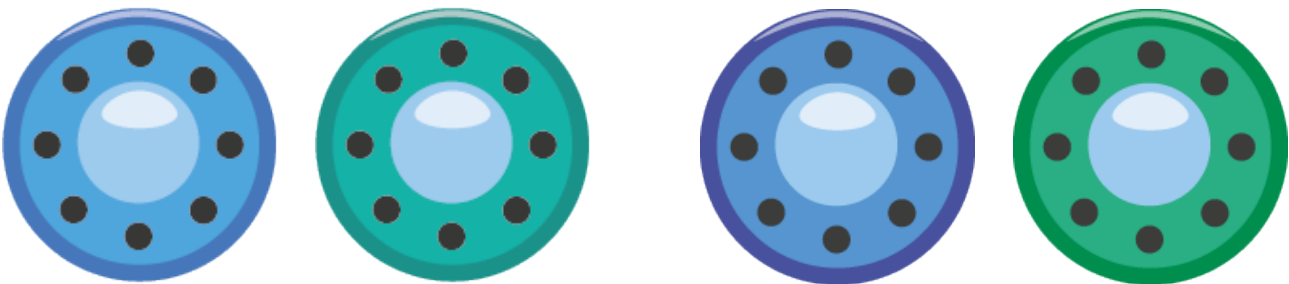


Imagen 46 - Naves azul y verde antes y después de modificar los colores

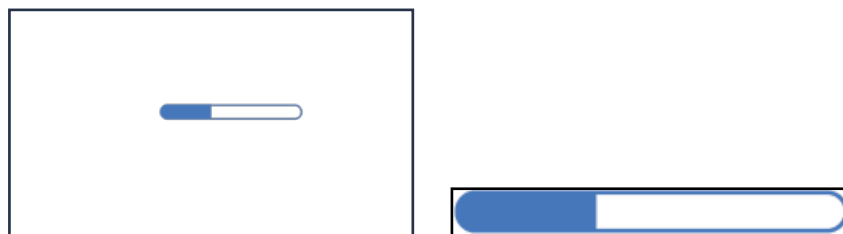


Imagen 47 - Animación barra azul antes y después de recortar el espacio sobrante



Imagen 48 - Animación nave+estrellas antes y después de recortar el fondo sobrante.



Imagen 49 - Animación estrella fugaz antes de modificarla



Imagen 50 - Recurso PNG sobre el que proyectar la animación estrella fugaz recortada



Imagen 51 - Secuencia de tres fotogramas de la animación estrella fugaz recortada.

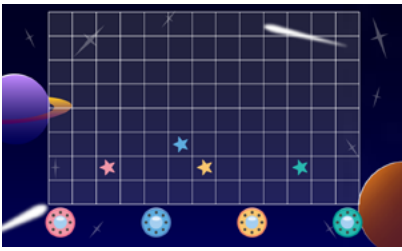


Imagen 52 - Nivel 1

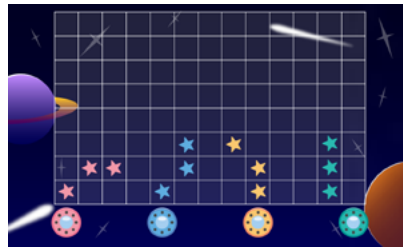


Imagen 56 - Nivel 5

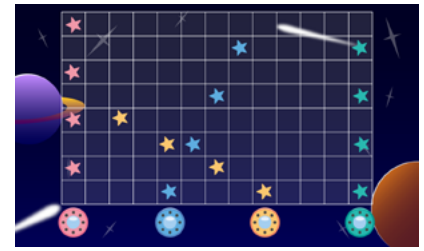


Imagen 60 - Nivel 9

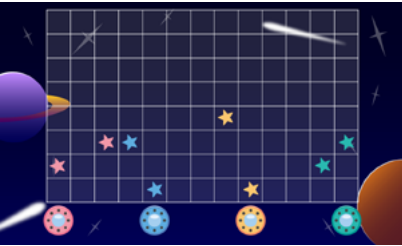


Imagen 53 - Nivel 2



Imagen 57 - Nivel 6



Imagen 61 - Nivel 10



Imagen 54 - Nivel 3



Imagen 58 - Nivel 7



Imagen 62 - Nivel 11



Imagen 55 - Nivel 4



Imagen 59 - Nivel 8



Imagen 63 - Nivel 12

PRUEBA 4

Solventados los problemas informáticos mencionados en la prueba anterior se realiza esta nueva prueba del prototipo para localizar los últimos detalles y comprobar el funcionamiento de las modificaciones anteriores. Los problemas encontrados se describen en la tabla.

Tabla 9 - Problemas encontrados en la prueba 4 y cambios realizados tras ésta

Problemas encontrados	Cambios realizados
En el nivel 3 y el 5 el jugador rosa tiene la misma distribución de estrellas	Modificar las distribuciones. No puede haber la misma distribución en dos niveles tan cercanos
No hay pantalla inicial (comienza directamente con “¿Quién quiere jugar?”)	Hacer pantalla de inicio del juego con el nombre del juego. Justo antes de la de “¿Quién quiere jugar?” (Ver Imagen 64)
No hay pantalla final de recompensa. El juego acaba muy brusco.	Hacer una pantalla final del juego (Vídeo final) (Ver Imagen 68)
Faltan los recursos de las pantallas laterales	Crear unas pantallas laterales para el inicio del juego y otras para el resto del juego (Ver Imágenes 64 a 67)

A continuación se muestran los recursos modificados.



Imagen 64 - Recursos inicio juego. Proyección lateral izquierda y derecha sin deformar, como se verá en la pared

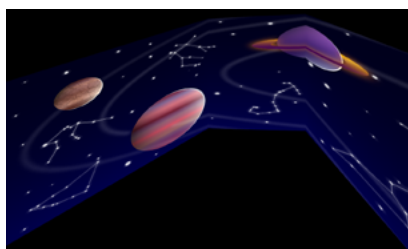


Imagen 65 - Recursos de las proyecciones lateral izquierda y derecha deformadas, listas para proyectar



Imagen 66 - Proyecciones lateral izquierda y derecha sin deformar, como se verá en la pared. Estas proyecciones se muestran durante todo el juego

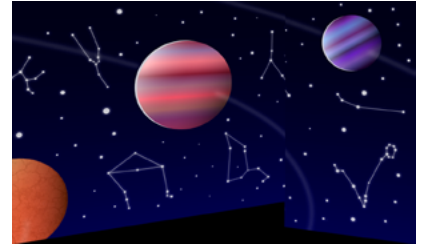
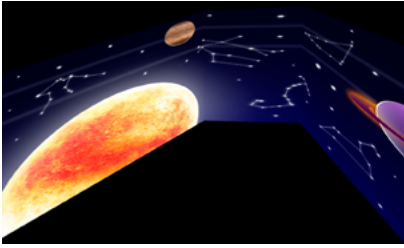


Imagen 67 - Proyecciones lateral izquierda y derecha deformadas, listas para proyectar

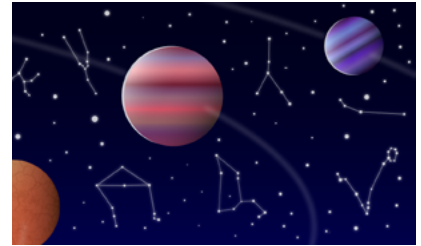
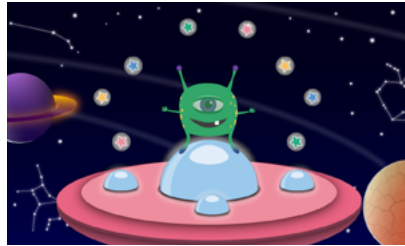


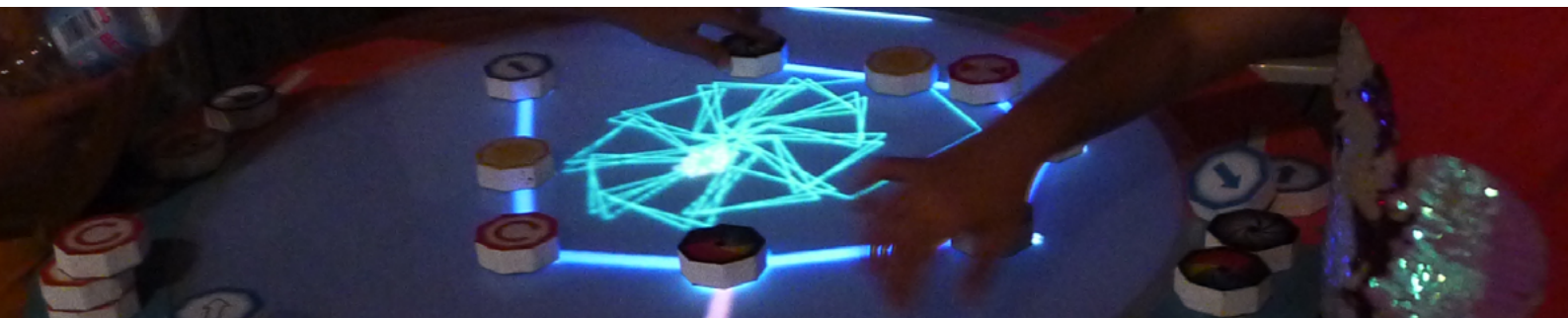
Imagen 68 - Recursos fin de juego. Proyección lateral izquierda y derecha sin deformar, como se verá en la pared y vídeo en la pared central

PRUEBA 5

Con esta prueba termina el proceso iterativo y la fase de desarrollo e implementación dando lugar a la fase de evaluación y resultados que se detalla en la sección 5.1.

El documento explicativo final del juego se muestra en el *Anexo H - Star Loop*, apartado *H.3. Documento explicativo final del juego* para aclarar y mostrar el juego final.

4.2. AMPLIACIÓN TURTAN



Cabe recordar que el juego TurTan ya existe y únicamente se pretende realizar una ampliación de la sintaxis del juego tras ver las posibilidades que este juego puede ofrecer a los usuarios para iniciarse de una forma divertida y dinámica en los conceptos de la programación.

En esta fase, se realizan los diseños gráficos de las fichas de juego, tanto de las ya existentes como de las nuevas fichas ideadas en la fase anterior. Recordemos que este trabajo de fin de grado no abarca la implementación informática de las instrucciones del juego. Esta labor la realiza el director del presente proyecto.

4.2.1. DISEÑO GRÁFICO DE LAS FICHAS

A continuación se muestran los diseños finales de las fichas realizadas para TurTan. Para ver opciones descartadas del diseño de las fichas ver el *Anexo I - Ampliación TurTan*.

Finalmente se opta por emplear un color para cada concepto creando un conjunto de fichas multicolor con una estética semejante para darle uniformidad al conjunto. Algunas de las características comunes de todas las fichas son los marcos de color, el centro blanco y las sombras y degradados que les aportan volumen.

Fichas de la sintaxis original de TurTan. (Ver Imágenes 69 a 74) Recordar que el efecto de las instrucciones originales de TurTan se explican en el *Anexo F - Propuestas basadas en Logo*, apartado *F.1. TurTan* se encuentra la sintaxis original de TurTan.



Imagen 69 - Ficha "Avanzar pintando"

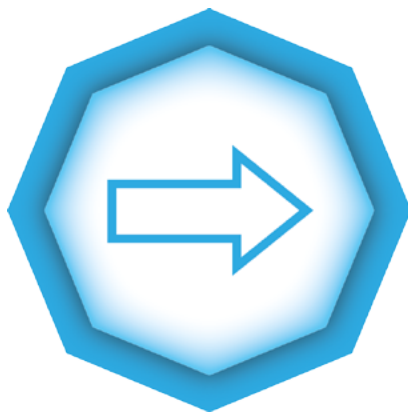


Imagen 70 - Ficha "Avanzar sin pintar"



Imagen 71 - Ficha "Girar"

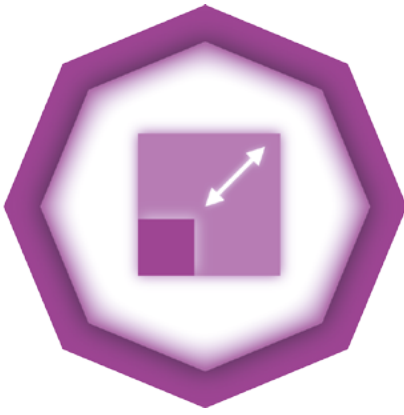


Imagen 72 - Ficha "Escalar"



Imagen 73 - Ficha "Volver al inicio"



Imagen 74 - Ficha "Repetir"

Fichas de la nueva sintaxis propuesta para TurTan. (Ver Imágenes 75 a 81) El efecto que provocan estas nuevas instrucciones propuestas ya se ha explicado anteriormente en el apartado 3.2.2. *JUEGO 2: Ampliación de TurTan para NIKVision redonda.*

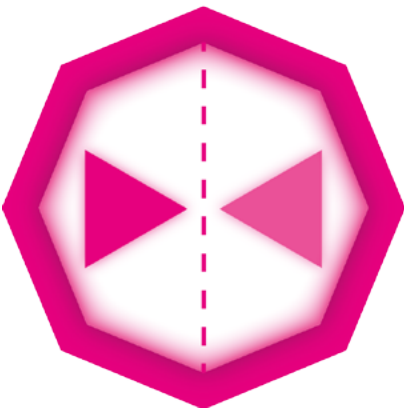


Imagen 75 - Ficha "Espejo radial (efecto kaleidoscopio)"



Imagen 76 - Ficha "Cambiar color"

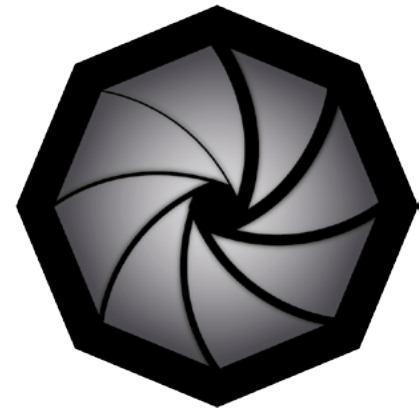


Imagen 77 - Ficha "Cambiar grosor"



Imagen 78 - Ficha "Polígono regular"



Imagen 79 - Ficha "Zoom"



Imagen 80 - Ficha "Animar"

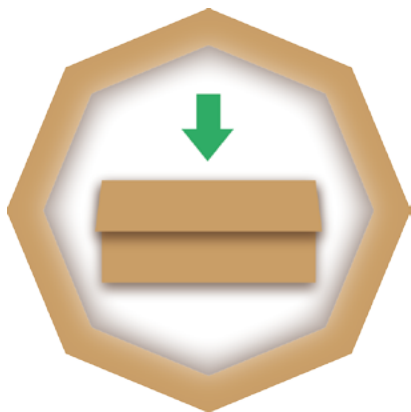


Imagen 81 - Ficha "Guardar"

En cuanto a la forma octogonal de las fichas se debe a una decisión de prototipado. Éstas se realizan en madera aglomerada y se cortan con la fresadora CNC del taller Open Art de Etopia, por ello tienen forma octogonal, para ser cortadas fácilmente y aprovechando la madera disponible. Una vez cortadas es necesario lijarlas y darles varias capas de imprimación hasta cubrir bien todos los poros. Después de lijan por última vez con una lija fina y se pegan todos los diseños gráficos anteriores y los fiduciales correspondientes. Para ver imágenes del proceso y más detalles ver el *Anexo I - Ampliación TurTan*, apartado *I.3. Prototipado de las fichas*.

A diferencia del juego Star Loop, este juego no necesita del desarrollo de recursos multimedia, ya que los gráficos consisten en gráficos vectoriales generados proceduralmente a partir de la manipulaciones de las fichas de juego sobre la mesa. Por tanto, una vez definida la sintaxis de las nuevas instrucciones añadidas a las originales de TurTan, y elaboradas las fichas, se puede proceder al desarrollo y codificación del prototipo funcional (en este caso realizado por el director de este trabajo). Una vez terminado el prototipo, se procede a la evaluación con niños, como se describe en la sección 5.2.

FASE DE EVALUACIÓN Y RESULTADOS

5

5.1. STAR LOOP

Se realiza una evaluación del juego final con usuarios objetivos, niños, con el fin de observar como se desarrolla el juego y ver si éste necesita grandes cambios.



Imagen 82 - Niños jugando durante la sesión

Para esta evaluación se cuenta con la participación de tres usuarios, una niña de 8 años (jugador 1) y dos niños de 10 (jugador 2) y 11 años (jugador 3). Para el niño de 10 años era la primera vez que realizaba una actividad relacionada con la programación, mientras que los niños de 8 y 11 años ya tenían alguna experiencia de programación con lenguaje Scratch, aunque para la niña de 8 años el manejo previo con Scratch había sido muy básico, sin incluir conceptos avanzados como bucles o procedimientos. (Ver Imagen 82)

Esta sesión de evaluación se graba para poder revisar, observar y analizar con detalle la experiencia vivida por los niños. Esta evaluación y su grabación permiten evaluar el diseño del juego y su adecuación al usuario objetivo. Para ello se coloca una cámara fija que graba a los niños y sus reacciones durante toda la sesión y con otra cámara se van captando diferentes momentos desde otras perspectivas (las mesas, las paredes, etc.).

Los niños resolvieron todos los niveles del juego y estuvieron jugando desde las 18:00 horas hasta las 19:00 horas. Al finalizar el juego, como recompensa por el tiempo dedicado se les compensa con chucherías.

En esta sesión de evaluación hay dos fuentes de información, que son las observaciones realizadas tanto en la propia sesión como con la visualización de los vídeos mencionados anteriormente y los comentarios y opiniones de los niños sobre el juego. Gracias a estas fuentes de información se pueden detectar los problemas del juego y proponer soluciones para éstos.

OBSERVACIONES

En estas observaciones se incluyen todos los detalles detectados durante la sesión y en la posterior revisión de los vídeos realizados. Estas observaciones nos permiten detectar los problemas de usabilidad que tiene el juego y elaborar una lista con éstos para poder reflexionar y proponer ideas que solucionen estos problemas detectados.

Las explicaciones de Spowi, parecen no ser suficientes en algunos casos, ya que hay detalles de las explicaciones que los jugadores no recuerdan durante el juego o de los que no se han enterado. Ésto puede deberse a que los jugadores necesiten escuchar más de una vez las explicaciones para reforzar las normas del juego. Pero en ocasiones faltan detalles que las instrucciones no mencionan. Por ejemplo, Spowi podría avisarles de que retiren la ficha de la nave de la zona de ejecución, éste es uno de los casos que más adelante se detallan por necesitar la intervención de un adulto.

A lo largo del juego es necesaria la intervención de los adultos en varias ocasiones que se detallan a continuación:

1. En la pantalla inicial de “¿Quieres jugar?”, aunque previamente se les ha dado la ficha de la nave para evitar que la tuviesen que buscar no llegan a tiempo a poner la ficha sobre la mesa. Se ha observado que el problema no está en que no supieran que tenían que poner la ficha en la mesa, sino que dudaron por el hecho de tener una nave de un color distinto al de la mesa en la que estaban (el jugador 2 tenía el mismo color de nave que el de la mesa y no dudó en que debía hacer). Por este motivo se ha necesitado intervención de los adultos en esta pantalla para aclarar lo que deben hacer y al iniciar de nuevo el juego que no se repita la misma situación.
2. Tras las explicaciones iniciales de Spowi los niños se quedan parados mirando sus mesas y intervienen los adultos para explicarles donde tienen las fichas y que deben hacer de forma muy resumida. Además, dado que tienen la ficha de la nave en la mesa se les explica la función de ésta y cómo funciona (ejecuta el programa al colocarla sobre la nave de la mesa), por lo que deben tener cuidado de no ponerla sobre la nave cuando no quieran que ejecute el programa.
3. En este primer nivel los jugadores 2 y 3 no necesitan más explicaciones, pero el jugador 1 sí. Inicialmente comienza a colocar las fichas en las casillas, pero sin tener en cuenta la función de las líneas. Entiende la mesa como una cuadrícula de casillas donde coloca las fichas recreando el recorrido que señalan éstas (ver Imagen 83). Se le explica que debe realizar la secuencia de instrucciones en la línea blanca y lo entiende, pero aun así le cuesta cuatro o cinco intentos superar el nivel, pero ya por fallos menores de observación y planificación del recorrido.
4. En los niveles más complicados donde deben usar todos los conceptos aprendidos ha sido necesaria la intervención de un adulto para ayudarles a visualizar el recorrido y planificarlo. Esta intervención ha consistido en realizar preguntas para ayudarles a ver la solución y plantear el recorrido en pequeños problemas individuales a resolver.
5. Es necesario recordarles en varias ocasiones que retiren la ficha de ejecutar de la nave de la mesa.

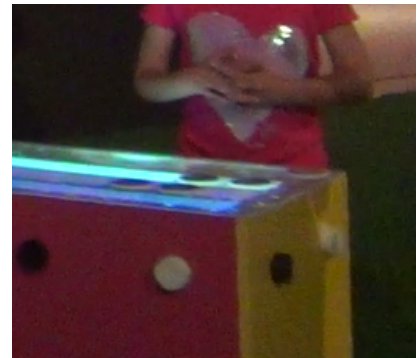


Imagen 83 - Primera interacción. Coloca las fichas sin tener en cuenta las líneas

Se dan otras situaciones que aunque no requieran la intervención de un adulto merece la pena destacar.

1. Destacar que cuando ya llevan unos pocos niveles superados comienzan a distraerse con otras cosas mientras sus programas se están ejecutando.
2. La primera reacción del jugador 1 con la ficha del bucle es intentar seleccionar de forma tangible el número de repeticiones con el dedo (sin girar la ficha), pero se da

cuenta por si mismo que debe girar la ficha (ver Imagen 84). Este es un ejemplo de que las explicaciones de Spowi no son suficientes o no logran captar la atención de los jugadores.

3. El juego se hace excesivamente largo y repetitivo, ya que el número de niveles es demasiado alto para lo rápido que entienden y aplican los conceptos.
4. Cuando un jugador resuelve un reto pero alguno de sus compañeros continúa intentándolo en lugar de ayudarlo se centra en buscar nuevas formas de resolver su programa.



Imagen 84 - Intento de seleccionar las repeticiones de forma táctil

Cabe destacar que el aprendizaje del jugador 1 ha sido muy rápido, ya que al principio no entiende bien en que consiste el juego y le cuesta superar los retos pero en el tercer nivel se pone al nivel de los jugadores 2 y 3. Los jugadores 2 y 3, los más mayores, lo hacen bien desde el primer nivel (salvo fallos puntuales).

En los niveles de bucle la mayoría de fallos son debidos a que cuentan mal el número de repeticiones, es decir con la visualización mental de lo que hacen las instrucciones cuando se complican las cosas. En definitiva, entienden el concepto del bucle muy bien incluyendo lo del espacio vacío entre fichas. A pesar de esto, en niveles más complicados les cuesta más visualizar y planificar la secuencia de instrucciones y tienen que enfrentar poco a poco el recorrido.

En los dos últimos niveles, donde deben aplicar todos los conceptos aprendidos, a los jugadores les cuesta más el hecho de visualizar el recorrido y decidir donde debe acabar el primer procedimiento y empezar el siguiente. Este punto de unión entre los dos procedimientos es el más problemático a la hora de resolver los retos. Hay que resaltar que en estos dos últimos niveles ya usan bucles sin problemas, tanto dentro como fuera de los procedimientos.

En cuanto a la cooperación entre los jugadores ha sido prácticamente nula. Apenas hay contacto visual entre ellos, cada uno se dedica a mirar su mesa y la proyección sobre las paredes. Tampoco han hablado entre ellos ni se han ayudado en momentos en los que alguno de ellos se atascaba o dudaba, únicamente en una ocasión el jugador 2 ha ido a la mesa del jugador 1 para resolver su programa. Cabe destacar que tampoco han solicitado ayuda a sus compañeros cuando no lograban resolver un reto.

La colaboración entre los jugadores del juego era un objetivo clave del proyecto ya que el espacio JUGUEMOS es colaborativo, pero plantear un reto compartido en una proyección compartida para todos no ha sido suficiente para que Star Loop sea colaborativo. Este hecho de que los jugadores no hayan colaborado puede deberse a que tanto en las instrucciones como en el transcurso del juego no se reúnen las consignas adecuadas para crear situaciones en las que realmente se necesite la cooperación para continuar jugando.

COMENTARIOS DE LOS NIÑOS

Otra fuente de información importante son los propios niños que tanto durante el juego como al finalizar éste los niños nos dieron sus opiniones que son las siguientes:

1. Tarda mucho en ejecutar
2. Podría haber una ficha que fuera ir más rápido
3. Hay muchos niveles
4. No les gusta tener que esperar a que sus compañeros acaben con sus estrellas

cuando ellos acaban antes.

Finalmente en esta sesión de evaluación y tras una reflexión concluimos que los problemas encontrados son los siguientes:

1. Son muchos niveles y se hace largo el juego
2. Se distraen y aburren durante la ejecución
3. No les da tiempo a colocar las fichas de las naves durante la cuenta atrás inicial
4. La ausencia de colaboración entre los jugadores

PROPUESTA DE SOLUCIONES

El aspecto más importante observado durante la evaluación es la ausencia de colaboración en el juego, ya que éste es uno de los objetivos del proyecto. Por ello, se le da mayor importancia que al resto y se plantean numerosas ideas para solventarlo. Para lograr un aprendizaje cooperativo se deben tener en cuenta cinco aspectos esenciales que son la interdependencia positiva, la exigibilidad personal, la interacción positiva cara a cara, las habilidades interpersonales y de grupo y el autoanálisis del grupo.

Aplicando estos conceptos se pueden provocar comportamientos colaborativos en los jugadores. Esto queda reflejado en las propuestas de mejora detalladas en el *Anexo J - Propuestas de mejoras para Star Loop*. Con las propuestas sugeridas se pueden crear más oportunidades de interacción en un contexto estructurado y solucionamos el problema de la nula comunicación haciendo legítima y necesaria la conducta que antes faltaba de pedir y proporcionar ayuda entre los individuos creándose situaciones en las que la única forma de alcanzar las metas grupales es a través de las metas personales de cada jugador.

Además, se proponen soluciones para los otros problemas menores de usabilidad encontrados y se detallan en el *Anexo J - Propuestas de mejoras para Star Loop*.

5.2. AMPLIACIÓN TURTAN

5.2.1. EVALUACIÓN 1

Se aprovecha la sesión de evaluación del juego Star Loop para que los tres niños que participan en la sesión prueben el juego TurTan.

DESARROLLO DE LA SESIÓN

Se les explica el funcionamiento del juego y la importancia de colocar las fichas en orden de izquierda a derecha para crear los programas. Además se hace referencia al modo de interacción de las fichas y se les enseña que todos los parámetros se modifican haciendo rotar las fichas.

Inicialmente se les explica las fichas más básicas (avanzar pintando, girar y repetir) y se les deja que experimenten y programen libremente. A la derecha se muestran unas imágenes del primer contacto con TurTan. (Ver Imágenes 85 y 86)

No tardan en probar nuevas fichas y mostrar interés por éstas, por ello se les va explicando poco a poco el funcionamiento de todas.

En algunas ocasiones, sobre todo al principio, les cuesta seguir la norma de colocar las nuevas fichas a la derecha de la anterior. Un ejemplo de este hecho se muestra en la imagen inferior. (Ver Imagen 87)

Además otro de los motivos por los que es necesaria la intervención de un adulto es para recordarles que hay que girar las fichas para modificar los parámetros de la instrucción y que esta provoque resultados visuales en la mesa.

Destacar que una de las fichas que más emoción les ha provocado ha sido la de “Guardar” porque así pueden guardar los dibujos más bonitos y enseñárselos a sus padres más tarde.

A continuación se muestran algunas imágenes de los programas realizados durante la sesión. (Ver Imágenes 88 a 92)

Por último, destacar que a pesar de que el juego TurTan no es colaborativo en sí mismo, ya que puede jugar un único jugador, las características físicas del tabletop redondo fomentan la interacción física y verbal entre los jugado-



Imagen 85 - Niños jugando a TurTan



Imagen 86 - Niños jugando a TurTan



Imagen 87 - A veces no colocan las nuevas fichas a la derecha de la anterior

res. Éstos manipulan físicas a la vez, sin guardar turnos, comentando entre ellos y sugiriendo cambios para crear nuevos gráficos. Por lo tanto gracias a las características físicas de la mesa los jugadores cooperan juntos para crear gráficos sorprendentes.



Imagen 88 - Niños interactuando con TurTan



Imagen 89 - Niños interactuando con TurTan



Imagen 90 - Niños interactuando con TurTan



Imagen 91 - Niños interactuando con TurTan

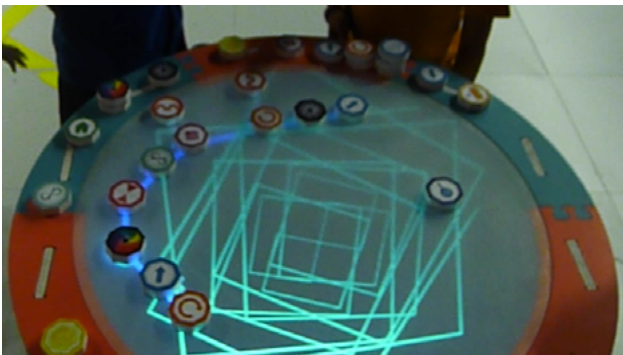


Imagen 92 - Niños interactuando con TurTan

5.2.2. EVALUACIÓN 2

Se realiza una segunda sesión en el colegio público Los Albares, de la Puebla de Alfinden, que se presta a ayudarnos en uno de sus últimos días lectivos. Contamos con la participación de dos clases de tercero de infantil, lo que se traduce en unos cincuenta niños de cinco años.

Para que la sesión se desarrolle con éxito es necesario transportar e instalar la mesa NIKVision redonda y todos los objetos y equipamiento necesarios. En el *Anexo I - Ampliación Tur-Tan* se encuentra la explicación de cómo se llevo a cabo todo este proceso previo a la sesión.

Esta sesión no se puede definir como una sesión de evaluación estricta, ya que los niños no tienen la edad de los usuarios objetivos planteada en las conclusiones de la fase analítica. Además, no se dan las condiciones necesarias para evaluar el juego correctamente, dado el elevado número de niños y el poco tiempo disponible (la sesión dura una hora), por lo que cada niño interacciona aproximadamente 4 minutos y lo hacen en grupos de tres. A pesar de todo ésto, se pueden extraer ciertas conclusiones sobre el diseño de las fichas, las nuevas instrucciones añadidas, de como el juego estimula la colaboración y de su usabilidad.

Las conclusiones obtenidas en base a las observaciones realizadas de la interacción con el juego y sus reacciones y opiniones sobre el juego y las fichas son las siguientes:

1. El tiempo de interacción de cada niño es muy breve, por lo que muchos no se acostumbran al modo de interacción y a las “normas” del juego.
2. Miran más a la proyección que a la mesa, ya sea por altura o por costumbre y hábito de juego.
3. Existe predilección por determinadas fichas en función del efecto que éstas crean en el juego (fichas de guardar y animar) o por su atractivo visual (ficha cambiar color).
4. El relevo constante de niños durante la sesión dificulta que las explicaciones y consejos lleguen a oídos de todos impidiendo así una interacción fluida y fructífera. A pesar de ésto se alcanzan programas sorprendentes.

A continuación se muestran algunas imágenes de esta sesión de evaluación. (Ver Imágenes 93 a 97)



Imagen 93 - Niños interactuando en la mesa y proyección en la pared



Imagen 94 - Niños interactuando en la mesa



Imagen 95 - Niños interactuando en la mesa



Imagen 96 - Niños interactuando en la mesa



Imagen 97 - Niños interactuando en la mesa

CONCLUSIONES Y TRABAJO FUTURO

6

6.1. CONCLUSIONES

Se han realizado dos actividades basadas en la interacción tangible para el espacio JUGUE-MOS. Star Loop es un juego de temática espacial en el que los niños aprenden conceptos de programación de forma progresiva y está basado en el modelo de programación por bloques de Scratch. Por otro lado, se ha realizado una ampliación de TurTan (desarrollado por Carles F. Julià, Daniel Gallardo y Sergi Jordà en la Universidad Pompeu Fabra de Barcelona) para el nuevo tabletop NIKVision redondo del grupo de investigación GIGA Affective Lab, con el objetivo de aportarle nuevas funciones y herramientas de programación con las que experimentar y crear nuevos resultados visuales.

Ambas actividades se han diseñado siguiendo una metodología clara y estructurada que ha permitido cumplir con la planificación establecida. Gracias a la fase de investigación y a la posterior fase de ideación y diseño se ha alcanzado el desarrollo del juego Star Loop, que difiere de los juegos ya existentes en el mercado referentes a lenguaje computacional y la ampliación de TurTan. Además, para el desarrollo de las actividades se han empleado diferentes software, como Adobe InDesign, Adobe Illustrator, Adobe Photoshop y Adobe Premiere, afianzando y perfeccionando su manejo.

Lo novedoso de estas actividades es que permiten jugar a varios niños a la vez y que se basan en el modelo de interacción tangible, lo que facilita un aprendizaje lúdico y colaborativo.

Gracias a la disponibilidad y ayuda de la becaria del proyecto Cesar y el director del trabajo, quienes han llevado a cabo el desarrollo informático de Star Loop y la ampliación de TurTan respectivamente, ha sido posible realizar un proceso de diseño iterativo que ha permitido mejorar y refinar los fallos encontrados en las iteraciones.

Finalmente han sido evaluadas ambas actividades con un grupo de tres niños del rango de edades definido previamente. Estas evaluaciones han permitido testar los juegos y sacar conclusiones reales. No se han detectado fallos importantes de usabilidad, puesto que todos los niños han sido capaces de seguir los juegos y se ha producido un proceso de aprendizaje de programación adecuado. Sin embargo, especialmente en Star Loop se echa en falta que se fomente la colaboración entre los jugadores, ya que se ha observado que los jugadores no interaccionan apenas entre ellos a pesar de encontrarse en un espacio colaborativo.

Se concluye que se puede mejorar aplicando una serie de conceptos y propuestas planteadas anteriormente, por lo que queda abierta la posibilidad de continuar trabajando en estas actividades.

6.2. TRABAJO FUTURO

En cuanto al futuro se plantean nuevas mejoras para Star Loop con la esperanza de alcanzar el objetivo de colaboración propuesto al inicio del proyecto para lograr un juego colaborativo donde los niños interaccionen entre ellos. Para conseguir ésto se plantean una lista de propuestas en el *Anexo J - Propuestas de mejoras para Star Loop* que pueden servir de inspiración y punto de partida para el trabajo futuro. Además, se deja abierta la opción de incluir nuevos conceptos que enseñar a los usuarios como por ejemplo expresiones condicionales y booleanos.

Por otro lado se propone continuar en la línea de trabajo de TurTan en la búsqueda de nuevas funciones y aplicaciones para éste, debido al gran interés y motivación que produce en los usuarios, tanto niños como adultos. Como foco de atracción resulta interesante la idea de incluir efectos sonoros sin sustituir su función principal que es la de crear gráficos visualmente muy atractivos ni crear una mesa de mezclas de sonidos.

BIBLIOGRAFÍA

- [1] Web del GIGA Affective Lab
<http://giga.cps.unizar.es/affectivelab/>
- [2] Marco, J., Cerezo, E., Baldassarri, S., Mazzone, E., & Read, J. C. (2009, September). Bringing tabletop technologies to kindergarten children. In Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology (pp. 103-111). British Computer Society.
- [3] Marco, J., Baldassarri, S., & Cerezo, E. (2013, February). ToyVision: a toolkit to support the creation of innovative board-games with tangible interaction. In Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (pp. 291-298). ACM.
- [4] Web del proyecto JUGUEMOS
<http://cesaretopia.com/juguemos/>
- [5] Web reacTIVision
<http://reactivision.sourceforge.net/>
- [6] Web de Ubisense
<https://ubisense.net/en/blog/manufacturing/everything-you-need-know-about-real-time-location-system-rtls>
- [7] Web de Microsoft
<https://developer.microsoft.com/es-es/windows/kinect>
- [8] CRATTY, B. J. (1989). Desarrollo perceptual y motor en los niños. Madrid. Paidós.
- [9] PIAGET, J. (1975). El mecanismo del desarrollo mental. Ed. Nacional. Madrid.
- [10] Johnson, D. W., Johnson, R. T. y Holubec, E. J. (1994). Cooperative Learning in the Classroom (p. 5). Association for Supervision and Curriculum Development (ASCD).
- [11] Galindo, F.C. & De la Varga, J.M. (2016). EL MÉTODO PUZZLE COMO TÉCNICA PARA EL APRENDIZAJE COOPERATIVO DE LOS MAPAS ESTRATÉGICOS: UNA EXPERIENCIA EN LA ASIGNATURA “ADMINISTRACIÓN DE ORGANIZACIONES” (p. 3). Universidad de Málaga (España).
- [12] Web MIT Media Lab, Illuminating clay
<http://tangible.media.mit.edu/project/illuminating-clay/>
- [13] Web MIT Media Lab, Topobo
<http://tangible.media.mit.edu/project/topobo/>
- [14] Web MIT Media Lab, I/O Brush
<http://tangible.media.mit.edu/project/io-brush/>
- [15] Web Superflux, Sketch a move
<http://superflux.in/index.php/work/sketch-a-move/#>

- [16] Web reactable
<http://reactable.com/>
- [17] Web MIT Media Lab, Siftables
<https://www.media.mit.edu/videos/labcast-20-siftables/>
- [18] Web Jonas Friedemann Heuer, Noteput
<http://www.jonasheuer.de/?noteput>
- [19] Web MIT Media Lab, SoundFORM
<http://tangible.media.mit.edu/project/soundform/>
- [20] Web imascono, Movistar Team Chapas
<http://imascono.com/es/portfolio-item/raspberry-pi-mesa-interactiva>
- [21] Definición del Pensamiento Computacional por la ISTE y la CSTA
<http://www.pensamientocomputacional.org/index.php/home/menu-definicion>
- [22] Web Club Jovenes Programadores de la Universidad de Valladolid
<https://scratch.infor.uva.es/cjp-uva/>
- [23] Entrevista a Miguel Ángel García Guerra
<http://videojuegosretro-upm.blogspot.com.es/2015/01/entrevista-miguel-angel-garcia-guerra.html?view=sidebar>
- [24] Web xataka, Elders of Madness (Videojuego creado por alumnos de secundaria)
<https://www.xataka.com/videojuegos/elders-of-madness-el-videojuego-creado-por-alumnos-de-secundaria-para-aprender-algebra-y-literatura>
- [25] Entrevista a Juan Julián Melero
<https://www.genbetadev.com/entrevistas/la-ensenanza-del-pensamiento-algoritmico-debe-empezar-en-primaria-entrevista-a-juan-julian-melero>
- [26] Web First LEGO League
<http://www.firstlegoleague.es/>
- [27] Web Code
<https://code.org/>
- [28] Web Genios, proyecto de Google y Ayuda en Acción
<http://www.genios.org/>
- [29] Web Blogthinkbig, Aprender a programar es el futuro de la educación primaria
<http://blogthinkbig.com/aprender-programar-futuro-educacion/>
- [30] Web Wikipedia, Seymour Papert
https://es.wikipedia.org/wiki/Seymour_Papert
- [31] Web MIT Media Lab, Fundación LOGO
<http://el.media.mit.edu/logo-foundation/>

- [32] Web Scratch del MIT Media Lab
<https://scratch.mit.edu/>
- [33] Web Wikipedia, Mitchel Resnick
https://es.wikipedia.org/wiki/Mitchel_Resnick
- [34] Web MIT Media Lab, Mitchel Resnick
<https://www.media.mit.edu/people/mres/overview/>
- [35] Web Tangible Interaction Design and Learning Laboratory, Strawbies
<http://tidal.northwestern.edu/blog/strawbies/>
- [36] Web Play Osmo
<https://www.playosmo.com/es/coding/>
- [37] Web Light-Bot
<https://lightbot.com/>
- [38] Web Cargo-Bot
<https://twolivesleft.com/CargoBot/>
- [39] Web Kodable
<https://www.kodable.com/>
- [40] Web KinderLab Robotics, KIBO
<http://kinderlabrobotics.com/kibo/>
- [41] Web Primotoys, Cubetto
<https://www.primotoys.com/es/>
- [42] Web Juguetrónica, Dash & Dot
<https://www.juguetronica.com/dash-dot>
- [43] Web BQ, Zowi
<https://www.bq.com/es/zowi>
- [44] Web Digital Dream Labs, Puzzlets
<http://www.digitaldreamlabs.com/>
- [45] Web Bee-Bot
<https://www.bee-bot.us/>
- [46] Web Bee-Bot, Blue-Bot
<https://www.bee-bot.us/bee-bot/bee-bot/bluebot.html>
- [47] Web Bee-Bot, Pro-Bot
<https://www.bee-bot.us/probot.html>
- [48] Gallardo, Daniel & F. Julia, Carles & Jordà, Sergi. (2008). TurTan: A tangible program-

ming language for creative exploration. 2008 IEEE International Workshop on Horizontal Interactive Human Computer System, TABLETOP 2008. 89 - 92. 10.1109/TABLETOP.2008.4660189.

- [49] Web Robot Turtles
<http://www.robotturtles.com/>
- [50] Web Code Monkey Island
<http://codemonkeyplanet.com/>

